An improved shuffled frog leaping algorithm and its application

Suonan Lengzhi[1,a] Yueguang Li [2,b]
[1]Gansu Normal University for Nationalities, Hezuo, gansu, 747000, China
[2]Gansu Normal University for Nationalities, Hezuo, gansu, 747000, China
[a]Yueguangli7@163.com, [b] Yueguangli7@sohu.com

Abstract

In order to enhance the performance of shuffled frog leaping algorithm in solving optimization problem, this paper added the mutation operator to original shuffled frog leaping algorithm, an improved shuffled frog-leaping algorithm of solving TSP was proposed. Simulation results show that the improved algorithm in the optimization accuracy, convergence speed and success rate had more than doubled to improve.

Keywords: Shuffled frog leaping algorithm; TSP; mutation operator; local search

0. Introduction

Shuffled frog leaping algorithm is proposed by Muzaffar Eusuff and Kevin Lansey on 2000 year, it is a swarm intelligence computation optimization algorithm [1]. It is used to solve discrete combinatorial optimization problem. As a new bionic intelligent optimization algorithm, it combined with memetic algorithms and particle swarm optimization algorithm, it has the advantages of two intelligence optimization algorithms. The features of the algorithm are concept simple, parameters is few, calculation speed is fast and global search ability is strong and easy to implement [2]. The main application of shuffled frog leaping algorithm is to solve multi objective optimization problems, such as water resources allocation, pier maintenance, workshop arranging flow engineering problem [3]. Domestic and foreign scholars have conducted a lot of research on it. Such as: Zhang joins "cognitive component" into the internal search strategy, improve the success rate of the algorithm for solving ability and jump out of local optimal solution [4]; Zhao Pengjun joins the attraction repulsion mechanism into the internal search strategy, effectively avoid the algorithm premature convergence[5]; Hatem E in the internal search strategy through the introduction of the "search accelerated factor", to improve the global searching ability of the algorithm[6].

Traveling salesman problem is a classical combinatorial optimization problem, which belongs to the NP-hard problems. It can be described as follows: Give a graph G = (V, E), E is the set of edges, nonnegative weights are w (E) on each side, how to find the Hamilton ring of G, making w (E) is minimum? Its practical meaning is: for a given K City, traveling salesman from a city of departure access the rest of the city not repeated returned the starting city finally, the requirements of all traveling salesmen to find a shortest route to travel. With the growing of the number of city, the solution space will grow exponentially, through the exhaustive method cannot solve, therefore optimization algorithm is used to solve the TSP is very necessary. Such as simulated annealing algorithm, ant colony algorithm, genetic algorithm, particle swarm algorithm.

1. The mathematical model of shuffled frog leaping algorithm

Shuffled frog leaping algorithm simulates the frog population when they are searching for food, according to ethnic classification of memes of information transfer. Shuffled frog leaping algorithm mainly includes two parts: local search and global information exchange. The following is a brief introduction to the mathematical model of shuffled frog leaping algorithm.

A randomly generated F frog consists of initial population, each frog expresses a feasible solution of problem, $U=\left(U^1,U^2,...,U^d\right)$, calculate the frog individual fitness $f(i)$, where *d* denotes the dimension of the solution space. After the random generation of initial population, the individuals frog according to the fitness $f(i)$ in descending order is stored in the $X = \{U(i), f(i), i = 1,...,F\}$, then according to the specific principle to divide the whole frog population into m groups $Y^1, Y^2, ..., Y^m$, each group contains the *n* frogs, satisfy the following relations:

$$Y^K = \left[U(i)^k, f(i)^k, \middle| U(i)^k = U(k + m(i-1)), f(i)^k = f(k + m(i-1)), i = 1,...,n\right] k = 1,...,m; F = mn \tag{1}$$

In the frog population, the aim of various group perform local search strategy is to search the local optima in the different search direction, after a certain number of iterations, making the local optimal individuals in the population tends to the global optimal individual.

First of all, the frog population is divided into a plurality of groups, local search is carried out for each ethnic group, in order to avoid the frog individual into a local optimum prematurely, while speeding up the convergence process, in each group, according to specific principles choose a certain number frogs constitute the ethnic sub family group. For the frog population, with global best fitness of the solution is expressed as U_g; for each sub groups, with the best fitness of the solution is expressed as UB, the worst fitness solutions expressed as UW. The local search is carried out for each sub population, update strategy as following:

$$S = \begin{cases} \min\{[rand(U_B - U_W)], S_{\max}\}, & U_B - U_W \geq 0 \\ \max\{rand(U_B - U_W), S_{\max}\}, & U_B - U_W < 0 \end{cases} \tag{2}$$

$$U_q = U_W + S \tag{3}$$

Among them, *S* expresses the adjustment vector of frog individual, S_{mac} represents the maximum step size of the frog allows to change.

Global information exchange helpful to collect local information of all kinds of group search, by meme transmission, obtain the search direction of new global optimal solution. After all populations conduct a certain number of local searches , various groups of frogs are mixed together, according to the fitness of $f(i)$ in descending order, the re division of population, which makes the meme information of the frog individual obtain the full transfer, then continue to conduct local search, so repeatedly until it is convergent, the algorithm stop.

2. Improvement shuffled frog leaping algorithm for solving TSP

Although the shuffled frog leaping algorithm in the literature [7] using mixed local search can guarantee the feasibility of the update solution, but a random intercept operation is blindness, the local search is easy to produce premature. Therefore, on the basis of the literature [8], this paper introduces the adjustment factor and the adjust order thought, at the same time, a mutation operation is added to the global information exchange process, so as to put forward an improved shuffled frog leaping algorithm for solving TSP problem.

2.1 Adjustment factor and adjustment of sequence

The solutions of *d* cities of TSP for $U = (U^i) \quad i = 1,2,...,d$, define the adjustment factor is $TO(i_1, i_2)$, before the U^{i_2} position insert U^{i_1}, then $U' = U = (i_1, i_2)$ is the new solution of U through the adjustment factor $TO(i_1, i_2)$ operating. For example, when U = (1, 3, 5, 2, 4), the adjustment factor for *TO* (4, 2), $U = U + TO$ (4, 2) = (1, 2, 3, 5, 4).

Orderly arrangement of one or more adjustment factor is the adjust order, denoted as *ST*, $ST = (TO_1, TO_2, ..., TO_n)$, Where $(TO_1, TO_2, ..., TO_n)$ is the adjustment factor, the order between them is meaningful. U_A, U_B are two different solutions, the adjust sequence of $ST(U_B \Theta U_A)$ expresses the adjusting sequence of adjusting U_A for U_B.

$$U_A = U_B + ST(U_B \Theta U_A) = U_A + (TO_1, TO_2, ..., TO_n) = [(U_A + TO_1) + TO_2] + ... + TO_n \quad \textbf{(4)}$$

In order to make the adjustment factor more in the adjustment sequence, increase the replacement diversity of frogs, thereby increasing the searching ability, avoid getting into local optimum, this paper does not require solution sequence preprocessing.

2.2 The selection of communication mode

The basic steps of improvement shuffled frog leaping algorithm for solving TSP problems are the following:

(1) The initialization parameter (the frog population number is m, the frog number is n in the population (total number of the frog F= (mn), the frog number is q in sub groups, and the frog update iterations);

(2) Randomly generate F initial feasible solution, and calculate the fitness of the individual frogs;

(3) The frog individual is divided into m groups according to fitness in descending order, construct the sub groups;

(4) Local search. Update the frog individuals for each population group in the sub population groups in accordance with the method of this paper;

(5) Each frog individual mutation, such as the generation of new individuals better than original individual will replace the frog original individual into frog populations, re calculating the degree of adaptation;

(6) Determine whether the algorithm meets convergence conditions, if satisfied, output the optimal path sequence; otherwise, update the global optimal solution, return to step (3).

3. The experimental simulation

This paper uses the Eil51 and Eil101 in TSPLIB to test algorithm. Parameter settings: the total number of the frog population F = 10n, the number of population m = 10, the number of frog in the sub group q= 2 n/3, the iteration number IT=q, the biggest adjustment factor number l_{max}= n/ 2 (where n denotes the number of city). By using the average distance and the average time, the performances of algorithm are evaluated, the stability evaluation of the algorithm is based on relative error. The experimental simulation is carried out 50 times the average statistical computer simulation for each TSP in the same experimental conditions, as shown in Table 1, which according to the different scale TSP problem, the first line is the experimental results of shuffled frog leaping algorithm, the second line is the experimental results of improved shuffled frog leaping algorithm. Table 2 gives the optimal solution, average distance, relative error and average running time of improved shuffled frog leaping algorithm, improved particle swarm optimization algorithm and genetic local search algorithm for solving *Ei*l51, the running environment is roughly same. The experimental results show that, in the case of each algorithm running time is the same, the improved shuffled frog leaping algorithm compared with other 2 kinds of optimization algorithms in solving large-scale TSP problems, the performance is more stable.

Table 1The test results of shuffled frog leaping algorithm and the improved shuffled frog leaping algorithm in different scale TSP

The examples of TSP	The known optimal value	Algorithm of optimal value	The average distance	The relative error/%	The average running time/s
*Ei*151	428.87	428.87	436.76	1.84	17.42
		428.87	430.66	0.42	22.36
*Ei*1101	629	655	673	6.99	28.38
		629	649	3.18	42.74

Table 2The test results of algorithms in solving the *Ei*l51 problem

Algorithms	Algorithm of optimal value	The average distance	The relative error/%	The average running time/s
Improved particle swarm optimization algorithm	436.77	440.78	2.77	—
Genetic local search algorithm	431.99	437.83	2.09	18.44
Improved shuffled frog leaping algorithm	428.87	430.66	0.42	22.36

4. Conclusions

This paper introduced the adjustment sequence of thought on the basis of literature [8], put forward an improved shuffled frog leaping algorithm for solving TSP problem. Simulation results show that improved shuffled frog leaping algorithm in solving the TSP problem has better search performance and robustness.

References

[1] EUSUFF M M, LANSEY K E. Water distribution network design using the shuffled frog leaping algorithm[A].World Water Congress[C].2001.

[2] LI Y H, ZHOU J Z, YANG J J, et al. Modified shuffled frog leaping algorithm based on threshold selection strategy[J]. Computer Engineering and Applications, 2007, 43(35):19-21.

[3] EUSUFF M M, LANSEY K E. Optimization of water distribution network design using the shuffled frog leaping algorithm[J]. J Water Resource Plan Manage, 2003, 129(3):10-25.

[4] Zhang Xuncai, Hu Xuemei. An improved shuffled frog leaping algorithm with cognitive behavior [C]. 2008.

[5] Zhao Pengjun, Liu Sanyang. The shuffled frog leaping algorithm in solving complex function optimization problems[J]. Application and Research of Computer. 2009, 26(7): 2435 – 2437.

[6]HATEM E, EMAD E, TAREK H, et al. Comparison of two evolutionary algorithms for optimization of bridge deck repairs[J]. Computer-Aided Civil and Infrastructure Engineering, 2006, 21:561-572.

[7] LUO X H, YANG Y, LI X. Solving TSP with shuffled frog-leaping algorithm [A]. The Eighth International Conference on Intelligent Systems Design and Applications, 2008.