The Application of Improved Harmony Search Algorithm for Solving Shortest Path Problems

Zhi-wang Jiang[1,2,3], Hong-xia Zhang[1]

[1]Hebei Finance University, Baoding, 071000 ,China

[2]Science and Technology Financial Key Laboratory of Hebei Province, Baoding, 071000,China

[3]Hebei province university financial intelligence application technology R & D Center, Baoding,071000,China

Keywords: Harmony search algorithm; Shortest path problem; Combinatorial optimization

Abstract. The shortest path problem is a classical problem in graph theory, it is also one of the classical problems in the field of combinatorial optimization. This paper analyzes the basic harmony search algorithm model, and points out its insufficiency in solving the shortest path problem, then modifies the harmony memory and perturbation method. The simulation experiment shows that the improved harmony search algorithm (IHSA) has faster global convergence speed, and can obtain more accurate optimal path. IHSA is an effective solution to solve the shortest path problem. Keywords: multi-objective bi-level programming, genetic algorithm, interpolation.

Introduction

The shortest path problem is a classical problem in graph theory. Its aim is to find a path with the minimum cost from the start point to the target point in the network. The shortest path problem is the basis of resource allocation problem, route design and analysis problem and so on [1]. It is similar to the job shop scheduling problem, the two distribution problem and so on, they all belong to the combinatorial optimization problems. To solve the shortest path problem has important applications in many fields. Such as, in the intelligent transportation system, the shortest path problem can be used for vehicle scheduling and vehicle navigation [2]; in computer communication and network, the shortest path problem can be used for routing [3].

The algorithms for solve the shortest path problem are mainly divided into traditional methods and artificial intelligence algorithms. The traditional methods includes Dijkstra algorithm, A* algorithm, etc. Since the 20th century, with the development of artificial intelligent technology, there are many intelligent optimization algorithms [4-6]. Genetic algorithm (GA), Simulated Annealing algorithm (SA), Particle Swarm Optimization algorithm (PSO), Tabu search algorithm (TSA) and Ant Colony algorithm were applied to solve the shortest path problem. The traditional methods are unable to meet the requirements of operational efficiency for high dynamic topology. Besides, they only can obtain the shortest path between two points. They cannot find a group of the shortest path or the second shortest path. Most artificial intelligent algorithms not only tend to have higher time efficiency in solving the shortest path problem, but also can obtain the good effect. Therefore, they have been used successfully to find the shortest route.

Harmony search algorithm (HSA) is a heuristic global optimization search algorithm, which simulates the musician improvisation process [7-9]. The algorithm has showed better performance than Genetic algorithm (GA) and Particle Swarm Optimization (PSO) algorithm in the some problems. This article uses the priority coding scheme. According to the priority values of nodes in harmony vector，the algorithm constructs a corresponding path. And through updating the harmony memory, it gets the shortest path. In addition, the algorithm improved the perturbation equations, which improve the efficiency of searching the shortest path.

Description of the shortest path problem

According to the graph theory, we set a graph G = (V, E), V is the vertex set, E is the edge set. There are n(n≥1) vertices in the graph. They are numbered from 1 to n. The weight is set to W, X represents the complete route. Therefore, the mathematical model for the shortest path problem can be described as:

$$f = \min \sum w_{i,j} A\left(V_{i,j}\right)$$

(1)

Where,

$$A\left(V_{i,j}\right) = \begin{cases} 1, V_{i,j} \in X \\ 0, V_{i,j} \notin X \end{cases}$$

(2)

The basic harmony search algorithm Interpolation function

Harmony search algorithm (HSA) is a novel intelligent optimization algorithm proposed by Geem et al. in 2001, which simulates the principle of generating harmony in music play [7].The harmony is a note that is played by every musician. When the notes generate at the same time, they can form the harmony. A beautiful music can be created when the harmony are played continuously. But in music creation, each musician will keep trying. They try to make their musical notes match with other instruments, then remember a few notes that is more matching and debug with other instruments. If there are no good results, they will try the other notes. Every musician choosing the right notes so repeatedly, until the music forms one of the most harmonious states.

The specific processes of HSA are described as follows:

Step1: Determines the objective function of optimization problem, constraint conditions and the basic parameters of HAS, such as the harmony memory (HM), harmony memory size (HMS), harmony memory consideration rate (HMCR), pitch adjusting rate (PAR), the maximum number of iterations.

Step2: Initialize the harmony memory (HM).

When initialize, the algorithm randomly generates HMS solutions and places them into the HM. The specific form of HM can be expressed as:

$$\begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_n^1 & \vdots & f\left(X^1\right) \\ x_1^2 & x_2^2 & \cdots & x_n^2 & \vdots & f\left(X^2\right) \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ x_1^{HMS} & x_2^{HMS} & \cdots & x_n^{HMS} & \vdots & f\left(X^{HMS}\right) \end{bmatrix}$$

(3)

Where $X^m = \left(x_1^m, x_2^m, \ldots, x_n^m\right)$ is the mth solution vector, x_i^m the component of X^m, $1 \le i \le n$, $1 \le m \le HMS$;the objective function is $f\left(X^m\right)$.

Step3: Generate new solutions. Every iteration the algorithm generates a new solution $X^{new} = \left(x_1^{new}, x_2^{new}, \ldots, x_n^{new}\right)$, and the component of new solution x_i^{new} can be generated by the following three mechanisms:

i) Keep some components of the solutions in HM. With a certain probability to keep some components of the solutions in HM, i.e. the probability of x_i^{new} selecting form $X_i = \left\{x_i^1, x_i^2, \ldots, x_i^{HMS}\right\}$ is HMCR.

ii) Generate the component of new solutions randomly. I.e. the probability of x_i^{new} selecting form outside the HM is 1-HMCR.

iii) Perturb the new solutions using the above methods. According to a certain probability (PAR) to perturb the component of new solution x_i^{new} .The principle of Perturbation is expressed as:

$$x'^{new}_i = x_i^{new} + 2 * u * rand - u$$

(4)

Where x_i^{new} x'^{new}_i and x'^{new}_i are the ith component of solution before and after perturbing, respectively.

Step4: Update the HM. Judge whether the new solution is better than the worst solution in HM. If it is, replace the worst solution by the new solution and find the worst memory in the library new solution.

Step5: Repeat Step 3, Step 4, until the termination condition is reached, then output the best solution.

Improved harmony search algorithm

To overcome the drawbacks of HSA, this paper puts forward an improved harmony search algorithm (IHSA).Its specific ideas are represented as follows:

In HSA, when to update the HM, the algorithm generates new solutions X^{new} from HM with the probability of HMCR. This causes uncertain and random search direction. In order to solve this problem, this paper introduces the idea of weight. Each solution has a corresponding weight in HM. We distribute the weight as follows: the current best solution in the HM obtains the maximum weight wb; the rest of the HMS-1solutions share the rest of the weights. The rest of each solution obtains the following weight:

$$w_r = \frac{1 - wb}{HMS - 1}$$ (5)

Where $wb = 0.5$, this can guarantee the IHSA will be search according to the direction of the best solution and have the ability to jump out of local optimal.

Perturb the new solutions X^{new} with the perturbation probability PAR. In HAS, bandwidth u is fixed. This can lead to the convergence speed and the robustness is not ideal. In order to solve this problem, this paper proposed an adaptive bandwidth u':

$$u' = bw \times (gn/NC)$$ (6)

Where gn is the current number of remaining iterations, NC is the maximum number of iterations.

At the beginning of updating the HM, IHSA has a bigger perturbation to x'^{new}_i.But with the number of iterations increasing, the perturbation for x'^{new}_i becomes small. Compared to the constant bandwidth u of HAS, the adaptive bandwidth u' of IHSA can accelerate the convergence speed.

The implement of the shortest path problem based on IHSA

In this paper, the IHSA is applied to solve the shortest path problem. The algorithm uses the priority coding scheme to set each variable values in the harmony vector which represent the priority value of the node. From the start node, the shortest path is constructed by choosing the maximum priority value of adjacent nodes.

(1) The fitness function

The quality of the harmony vector is judged by the value of the objective function, and the global optimal solution is obtained by minimizing or maximizing the value of the function. In this paper, the target is to find the minimum value path from the start node to the target node. Therefore, the objective function is defined as follow:

$$f = \left(\sum_{j=1}^{N'-1} \cos t_{p,p+1} \right)^{-1}$$ (7)

Where vk is the set of the nodes which are on the shortest path. N' is the number of nodes in the set vk. $\cos t_{p,p+1}$ represents the cost of neighboring nodes in vk. When the fitness function value reaches the maximum value, the shortest path will be obtained. When the path which the harmony vector represents is an invalid path, the fitness will be given a penalty value 0.

(2) The process of IHSA

i) Initialized the HM randomly;

ii)   Calculating the fitness function value of each harmony vector in harmony memory. If a valid path is feedback, calculating the fitness function value according to Eq. (7); otherwise, the fitness will be given a penalty value 0.

iii)   A new harmony vector is produced by Esq. (4), (5) and (6). Comparing the new harmony vector with the worst harmony vector which is in the harmony memory, if the new one is better than the worst one, then replace the worst vector.

iv) If $N = N_{\max}$, stop the iterative process, then output the best harmony vector in the harmony memory. Otherwise, return Step 3).

Experimental simulation

In order to verify the validity of the algorithms for different network topologies, randomly generated network topology with 20-100 nodes, and in the range [0,100], the integer values are randomly selected as the cost of each vector. Set $N_{\max} = 500$, compare the simulation accuracy of the IHSA, PSO and HAS in Fig.1.We can concluded that the performance of the IHSA is better than PSO and HAS.
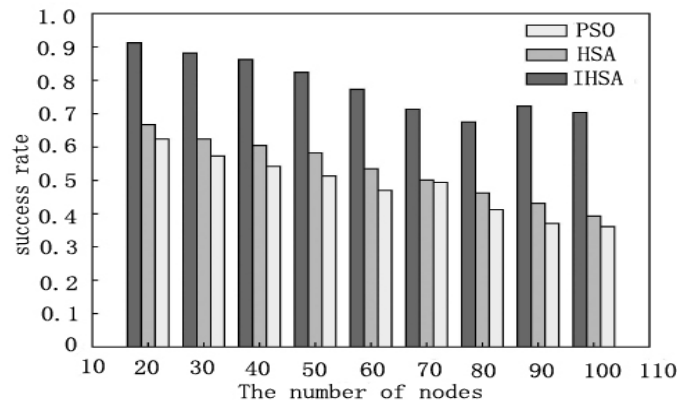

Fig.1: Comparison of route success rate between IHSA and PSO, HSA for networks of varying topologies

In the fig.2, it shows that the Standard deviation of the simulation results of IHSA, HSA and PSO algorithm. We can conclude that the Standard deviation of IHSA is lower than others.
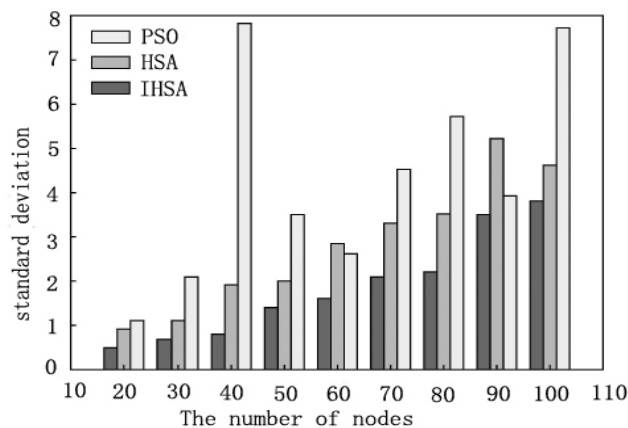

Fig.2: Comparison of standard deviation between IHSA and PSO, HSA for networks of varying topologies

In the fig.3, it shows us that average time of the simulation process. According to the figure.3, IHSA not only improves the success rate of the shortest path searching also ensure the operating efficiency of the algorithm.
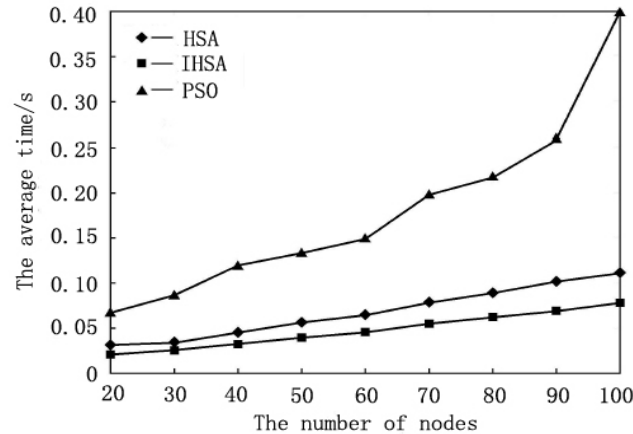

Fig.3: Comparison of convergence time between IHSA and PSO, HSA for networks of varying topologies

Conclusions

In this paper, IHSA is applied to the shortest path problem. The steps of HS is described in the paper and analyzing the limitation about HS based on the fact of the shortest path problem, then we put forward an improved HS to solve the shortest path problem. This paper adopts three kinds of performance indicators to compare the performance of IHSA, HAS and PSO algorithm in the shortest path problem. According to the simulation results, IHSA has a better performance than HAS and PSO. Therefore, IHSA can be applied to the shortest path problem.

Acknowledgements

References

[1] Zhi-xin Sun, Yan-juan Gao, Wen-ding Wang, Complete dynamic algorithm for updating shortest path tree. *Journal of Jilin University (Engineering and Technology Edition)*, 37 (6), pp. 860-864, 2007.

[2] Parichart P, Dongjoo P, Laurence R R, et al, Dynamic and stochastic shortest path in transportation networks with two components of t ravel time uncertainty. Transportation Research Part C: Emerging Technologies, 11 (4), pp. 331- 354, 2003.

[3] Desaulniers G, Soumis F, An efficient algorithm to find a shortest path for a car-like robot. *IEEE Transactions on Robotics and Automation,* 11 (10), pp. 819- 828, 1995.

[4] Manuel L, Barnes J W, Glover F, Intelligent scheduling with Tabu search: an applicatiom to jobs with linear delay penalties and sequence-dependent setup costs and times. *Applied Intelligence*, 3 (11), pp. 159-172, 1993.

[5] Wen-qi Huang, Duan-bing Chen, Ru-chu Xu, A new heuristic algorithm for rectangle packing. *Computers & Operation Research*, 34 (7), pp. 3270-3280, 2007.

[6] Mohemmed A W, Sahoo N C, Geok T K, Solving shortest path problem using particle swarm optimization. *Applied soft Computing*, 8 (12), pp. 1643- 1653, 2008.

[7] Zong Woo Geem, Kim J H, Loganathan G V, A new heuristic optimization algorithm: harmony search. *Simulatio,* 76 (8), pp. 60-68, 2001.

[8] XU Qiong, CHEN Rong-Qing, GUAN Yun-lan, TAO Guo-qiang, The Shortest Path Analysis Based on Genetic Algorithms. *East China Institute of Technology*, 26(2),pp:168-172,2003.

[9] Mohemmed A W, Sahoo N C, Geok T K, Solving shortest path problem using particle swarm optimization. *Applied soft Computing*, 8(4) ,pp: 1643- 1653, 2008.