Intuitionistic Fuzzy Hybrid Discrete Particle Swarm Optimization for Solving Travelling Salesman Problem

Hai-Tao Mei, Ji-Xue Hua, Yi Wang, Tong Wen

(Air and Missile Defense College，Air Force Engineering University，Xi'an 710051，China)

meiht14@163.com, huajx@163.com, Wangy@163.com

Keywords: intuitionistic fuzzy set, membership degree function, identical factor, the local optima, elitist reserving

Abstract. An intuitionistic fuzzy hybrid discrete particle swarm optimization (IF-HDPSO) is proposed for solving travelling salesman problem (TSP). By defining intuitionistic fuzzy charisma function, the IF-HDPSO algorithm exploits some other individuals to participate in the updates of velocity and position except the best one. In addition, the PSO identical factor function is defined to adjust inertia weight and learning operator adaptively, so the IF-HDPSO can explore the whole optimal solution quickly. Furthermore, an adaptive genetic algorithm based on elitist reserving strategy is developed, and combine it with PSO to reduce the probability of being trapped in the local optima and premature convergence. The simulation results indicate IF-HDPSO perform better on precision, iteration number and computational robustness.

Introduction

TSP is a famous combinatorial optimization problem, which is a NP-hard problem and cannot be solved within specific time scale by any known algorithm. Traditional algorithms existed for TSP such as stage dynamic programming, linear programming solution, and greedy method will get into the "combinational explosion problem" with the increase of city number. In recent years, people proposed many intelligence optimization algorithms, such as GA, ACO [1], PSO, firefly algorithm (FA) [2], and cuckoo search algorithm (CSA) [3]. Though these algorithms cannot ensure to find the best solution, they decrease the solution space and increase the probability of explores the optima.

The standard PSO was firstly proposed to solve the continuous space combinatorial optimization problems, as many optimization problems are defined in a discrete space, e.g. TSP and 0-1 knapsack problem. People proposed many improved PSO to solve the discrete problems. N. Salmani Niasar proposed a discrete fuzzy particle swarm optimization [4]. Except the best particle of the swarm; some other particles will be selected to participate in updating according to their degree of charisma.

In this paper, we propose an novel algorithm (IF-HDPSO) for solving TSP. We evaluate each particle's charisma with the intuitionistic fuzzy fitness, and define identical factor to adjust the parameters of PSO, including inertia weight and learning operator. Furthermore, the hybrid algorithm of an adaptive GA based on elitist reserving strategy and PSO is used to search the optima.

The rest of this paper is organized as follows. Section 2 briefly introduces TSP. Section 3 describes the proposed algorithm. Section 4 presents the result of a set of benchmarks of TSP from the TSPLIB in detail. Finally, Section 5 concludes with some discussions.

Representation of TSP

The representation of TSP is simple, assuming a TSP that is represented as $c=\{c_1,c_2,\cdots,c_n\}$, where n is the number of city, $c_i(i=1,2,\cdots,n)$ is the i-th visited directed edges in turn, the objective is to find a shortest path which visits each city exactly once [5], where $d(c_i,c_j)$ means the distance between c_i and c_j ($c_i,c_j \in c$). So, the cost of a permutation can be defined as follows：

$$\min f = \sum_{i=1}^{n-1} d(c_i, c_{i+1}) + d(c_n, c_1) \quad (1)$$

Intuitionistic Fuzzy Hybrid Discrete PSO

Discrete fuzzy PSO (D-FPSO) [4] which differ from the standard PSO is that the D-FPSO allowed some other particles, not just the best one, to participate in the updating of position and velocity based on their fuzzy charisma. Where the fuzzy charisma $\psi(h)$ is a fuzzy variable, and represented by the Cauchy membership function as formula (2).

$$Cauchy(x;\alpha,\beta) = 1 \Big/ [1+(\frac{x-\alpha}{\beta})^2] \quad (2)$$

Therefore, the position and velocity formula of D-FPSO should be modified as:

$$v_{id}(t+1) = \omega v_{id}(t) + c_1 r_1\ (p_{id} - x_{id}(t)) + \sum\nolimits_{h\in B(i,k)} c_2 r_2\, MF(h)(p_{gd} - x_{id}(t)) \quad (3)$$

Where $B(i,k)$ means a set of k best particles in the neighborhood of *i*-th particle, $MF(h)$ represents the fuzzy membership function of particle *h* in $B(i,k)$. The D-FPSO can solve the discrete TSP in some point, however, no specific methods can confirm the value of *k*, which need rich experiences and simulations, and it also will get into the local optima and premature convergence easily.

On the basis of D-FPSO[4], this paper proposes a hybrid discrete PSO (IF-HDPSO), we figure up the intuitionistic fuzzy membership in the neighborhoods of each particle and the intuitionistic fuzzy distance with the best particle of the whole swarm. Then, the distance which is less than the average of particle swarm will be allowed to enter into $B(i,k)$, and we defined an identical factor function of the swarm to adjust the inertia weight and learning operator adaptively, which can accelerate convergence speed of IF-HDPSO.

A. Intuitionistic fuzzy discrete PSO

Assuming a TSP instance, the objective function's fitness of particle *i* is represented as $f(x_i^t)$, where $f_{\max}(x_{id}^t)$ and $f_{\min}(x_{id}^t)$ mean the max and min fitness of *i*-th iteration. We estimate intuitionistic fuzzy charisma of particle *i* by the distance with the best one, the intuitionistic membership and nonmembership function of particle *i* are described as follows:

$$\mu(x_i^t) = \begin{cases} 0, & f(x_i^t) < f_{\min}(x_{id}^t) \\ \dfrac{f(x_i^t) - f_{\min}(x_{id}^t)}{f_{\max}(x_{id}^t) - f_{\min}(x_{id}^t)}, & f_{\min}(x_{id}^t) < f(x_i^t) < f_{\max}(x_{id}^t) \\ 1, & f_{\max}(x_{id}^t) < f(x_i^t) \end{cases} \quad \gamma(x_i^t) = \begin{cases} 0, & f_{\max}(x_{id}^t) < f(x_i^t) \\ \dfrac{f_{\max}(x_{id}^t) - f(x_i^t)}{f_{\max}(x_{id}^t) - f_{\min}(x_{id}^t)}, & f_{\min}(x_{id}^t) < f(x_i^t) < f_{\max}(x_{id}^t) \\ 1, & f(x_i^t) < f_{\min}(x_{id}^t) \end{cases} \quad (4)$$

The intuitionistic fuzzy charisma of particle *i* in *t*-th iteration can be described as an intuitionistic fuzzy number, for example $A = \{< x, \mu(x_{id}^t), \gamma(x_{id}^t) >\}$, and the best particle of the swarm can be described as $G = \{< p_g, 1, 0 >\}$ [6], the distance between particle *i* and p_g is d_{ig}.

$$d_{ig} = \sqrt{\frac{1}{3n}\sum_{i=1}^{n}[(\mu(x_{id}^t)-1)^2 + (\gamma(x_{id}^t)-0)^2 + (1-\mu(x_{id}^t)-\gamma(x_{id}^t))^2]} \quad (5)$$

The average distance of the whole swarm at iteration *t* to p_g is $\bar{d_i}$.

$$\bar{d_i} = \frac{1}{n}\sum_{i=1}^{n} d_{ig} \quad (6)$$

k is a dynamic variable in set of $B(i,k)$, which means the *k* shortest particles set.

$$\begin{cases} d_{ig} \in B(i,k), & if\ d_{ig} \le \bar{d_i} \\ d_{ig} \notin B(i,k), & else \end{cases} \quad (7)$$

Therefore, the velocity will be modified as below.

$$v_{id}(t+1) = \omega v_{id}(t) + c_1 r_1\ (p_{id} - x_{id}(t)) + \sum\nolimits_{h\in B(i,k)} c_2 r_2\, IMF(h)(p_{gd} - x_{id}(t)) \quad (8)$$

Where *IMF(h)* is the intuitionistic fuzzy fitness of particle *h* in set *B(i,k)*, and it can be calculated by Eq. (4).

B. Self-adaptive PSO algorithm

In PSO, the inertia weight can affect convergence speed and solution precision greatly [7], in early stage, we should increase the value of ω to enhance the global search ability. Other hands, in the later period of PSO, we should decrease ω to enhance the local search ability. c_1 and c_2 are two so called parameters to weigh the importance of self-cognitive and social-influence, so we should take bigger c_2 and smaller c_1 in early stage, which can enhance the diversity of the swarm. On the contrary, we should take bigger c_1 and smaller c_2 to enhance the local search ability and the convergence speed in the later period.

In the early stage of PSO, there are fewer identical individuals and the gap of fitness cannot be ignored. Adversely, identical individuals get increased with the process of iteration, we define identical factor s_v to describe the diversity of particle swarm, which can adjust the inertial weight adaptively.

$$s_v = 1 \bigg/ \left(1 + \frac{a}{N}\sqrt{\sum_{i=1}^{N} f_i^t - f_{avg}^t}\right) \tag{9}$$

In formula (9), where N is the population scale, Obviously, $s_v \in (0,1)$, f_i^t and f_{avg}^t represent the fitness of particle i of t-th iteration and average fitness of the whole swarm. $a = 0.1 \times N$ is modulation parameter. Afterwards, we define identical factor function $\theta_{s_v} = 1 - \cos(\frac{\pi}{2} s_v)$

Therefore, the adaptively parameter ω, c_1 and c_2 can be modified.

$$\omega^{t+1} = \omega_{\max} - (\omega_{\max} - \omega_{\min}) \cdot \theta_{s_v} \qquad c_1^{t+1} = c_1^{\max} - (c_1^{\max} - c_1^{\min}) \cdot \theta_{s_v} \qquad c_2^{t+1} = c_2^{\min} - (c_2^{\max} - c_2^{\min}) \cdot \theta_{s_v} \tag{10}$$

Where $\omega \in [0.4, 0.9]$, $c \in [0.5, 2.5]$, we take C-TSP instance as an example, the city path solution searched by standard PSO and self-adaptively PSO is illustrated in Figure 1.

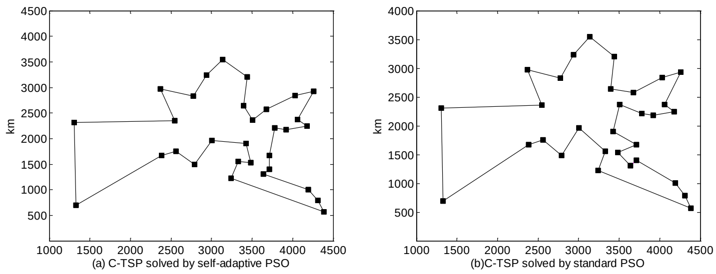

Fig.1. The comparison between self-adaptive PSO with standard PSO

C. Hybrid PSO algorithm

Like other intelligence optimization algorithm, PSO also suffers the disadvantages of premature convergence and get trapped into local optima, thought it can be realized easily and has high convergence speed. To enhance the standard PSO's ability of jump out the local optima, we propose an adaptive GA based on elitist reserving strategy, and combined it with PSO algorithm to solve the discrete TSP.

Procedures of GA mainly include selection, crossover and mutation, which are important methods to keep elitist and engender new individuals. We bring GA procedures into PSO, reserved the elitist individuals of the swarm and take p_{gd}, p_{id} as the parent individuals, then, we employ the adaptive selection, crossover and mutation operator to update p_{gd}, p_{id} by comparing the fitness of GA procedures around. This strategy can duplicate the excellent individuals and reject the worst particle.

Selection: Firstly, we reserve the best-so-far solution obtained by the whole swarm, and stores it in the set of crossover. Then, selecting other parent individuals by roulette wheel strategy, the selected probability is p_{x_i} .

$$p_{x_i} = f(x_i^t) \Big/ \sum_{i=1}^{n} f(x_i^t) \tag{11}$$

Crossover: p_c can affect the performance of GA directly. If crossover probability is very large, the excellent gene of parent individuals will be destroyed, and it will lead to convergent slowly. We perform crossover with adaptive probability.

$$p_c = p_{c\min} + (p_{c\max} - p_{c\min})\frac{f(x_i^t) - f_{\min}(x_{id}^t)}{f_{\max}(x_{id}^t) - f_{\min}(x_{id}^t)} = p_{c\min} + (p_{c\max} - p_{c\min})\mu(x_i^t) \tag{12}$$

Where $p_{c\min}$ and $p_{c\max}$ are the minimum and maximum crossover probability.

Mutation: Because the result of mutation operator is unknown, an appropriate mutation probability can increase the diversity of swarm and avoid being trapped in the local optima. We perform mutation with adaptive probability.

$$p_m = p_{m\min} + (p_{m\max} - p_{m\min})\mu(x_i^t) \tag{13}$$

Elitist reserving strategy: Elitist reserving strategy can store p_{gd} and take it as the parent individuals directly. Then, replacing the worst individual obtained by GA with p_{gd} after the crossover and mutation operation.

Experimental Results

The proposed IF-HDPSO algorithm is tested by some instances of TSP taken from the publicly available electronic library TSPLIB of TSP problems [8]. Most of the instances in the TSPLIB had already been solved by other algorithms and their optimal can be used to compare the performance of novelty algorithms. The comparison among GA, PSO, ACO, SA and IF-HDPSO are carried out.

We have set the parameter of proposed algorithm as, the population scale is 200 and iteration number is 300. Where basic parameter $\omega \in [0.4, 0.9]$ and $c \in [0.5, 2.5]$. The operator of GA, we take $p_c \in [0.4, 0.9]$ and $p_m \in [0.01, 0.1]$. For the justice and corrective of the experiment, we set the same test environment of GA, PSO, ACO, SA with IF-HDPSO, some other parameters are listed in Table 1.

Table 1 Parameter settings for GA, PSO, ACO and SA

Algorithm	Parameter settings
GA	$p_c = 0.8$, $p_m = 0.02$
PSO	$\omega = 0.75$, $c_1 = c_2 = 1.95$, $v \in [-0.5, 0.5]$
ACO	$\alpha = 3$, $\beta = 2$, $\rho = 0.3$, $Q = 5$
SA	$T_0 = 1000$, $T_{end} = 0.001$, $q = 0.9$, $L = 200$

In order to test the performance of IF-HDPSO algorithm with different city scale, we download 10 TSP instances from TSPLIB and test each problem for 20 runs. Table 2 lists the instance name, optimal solution in TSPLIB and the comparison result of 5 algorithms. The figure in the name of an instance represents the number of provided cities. For example, att48 provides 48 cities with their coordinates. Figure 3 shows the path planning of att48, eil51, eil101, ch150 and a280 searched by IF-HDPSO.

Table 2 comparison of experiments results of IF-HDPSO with other 4 algorithms

Instance	Optimal in TSPLIN		IF-HDPSO	GA	PSO	ACO	SA
bays29	9 074	Iteration number	18	37	29	40	33
		Result	9 074	9 074	9 074	9 178	9 074
		Deviation rate (%)	0	0	0	1.14	0
att48	33 522	Iteration number	25	54	41	37	56
		Result	33 522	33 966	33 784	33 522	34 623
		Deviation rate (%)	0	1.32	0.78	0	3.28
eil51	426	Iteration number	31	51	49	62	57
		Result	426	437	429	442	426
		Deviation rate (%)	0	2.58	0.70	3.75	0
pr76	108 159	Iteration number	39	79	68	80	67
		Result	108 159	114 548	110 140	108 354	110 197
		Deviation rate (%)	0	5.90	1.83	0.18	1.88
krob100	22 141	Iteration number	46	95	86	88	101
		Result	22 141	22 508	22 369	23 584	22 196
		Deviation rate (%)	0	1.65	1.02	6.51	0.25
gr120	6 942	Iteration number	61	106	96	115	121
		Result	6 942	7 083	7 009	6 984	7 049
		Deviation rate (%)	0	2.03	0.96	0.60	1.54
ch130	6 111	Iteration number	77	116	103	137	142
		Result	6 143	6 506	6 127	6 148	6 129
		Deviation rate (%)	0.52	6.46	0.26	0.61	0.29
ch150	6 528	Iteration number	92	122	125	159	154
		Result	6 528	6 842	6 867	6 665	6 547
		Deviation rate (%)	0	4.81	5.19	2.01	0.29
pr226	80 369	Iteration number	133	168	179	183	169
		Result	80 369	85 254	82 913	83 983	81 768
		Deviation rate (%)	0	6.01	3.17	4.50	1.74
a280	2 579	Iteration number	157	172	192	188	196
		Result	2 579	2 815	2 668	2 662	2 606
		Deviation rate (%)	0.19	9.15	3.45	3.22	1.05

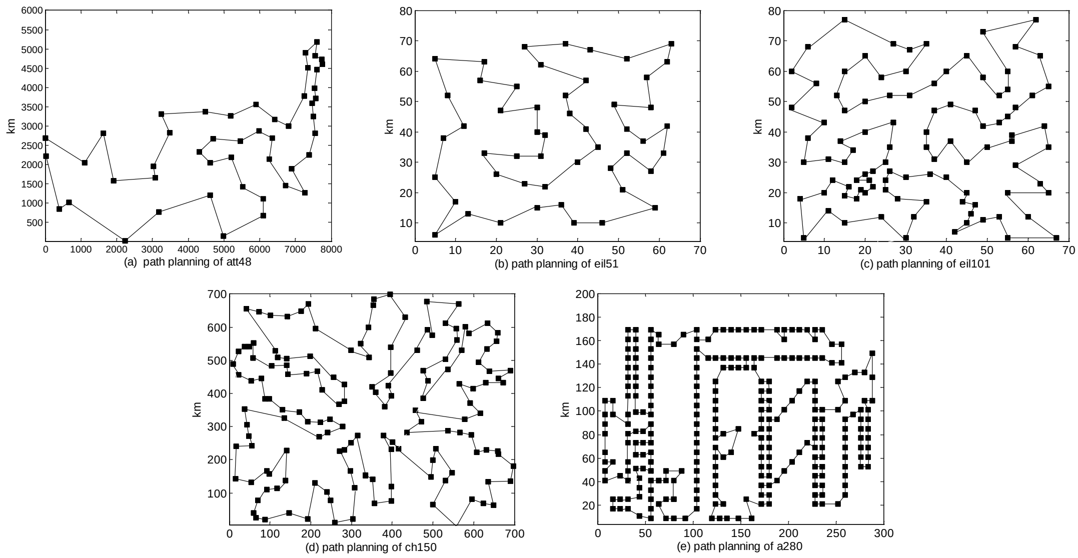

Fig. 2 Path planning of 5 TSP instances by IF-HDPSO

In Table 2, the proposed IF-HDPSO performs better in the iteration number and optimal solution compared with other 4 algorithms. Especially, for the bays29, 5 algorithms all can search the global optimal, the superiority of IF-HDPSO is not remarkable, but it needs fewer iteration number obviously. After analysis, the reason indicates IF-HDPSO algorithm adopts identical factor function to adjust inertia weight and learning factor, moreover, the GA based on elitist reserving strategy hybrid PSO can avoid IF-HDPSO being trapped in local optima and premature convergence, and guide the fly experience of particle to the global optima. Except ch130，it can get the known best solution of TSPLIB, and stabilize at the solution for many times. The path planning searched by other algorithms have many crosses, which means these algorithm is trapped into the local optima and stopped searching. Generally, for the complicated and discrete TSP instances, ACO and SA possess quicker convergence speed compared with GA and PSO, and they can seek out the better path planning, however, when the dimension is greater than 100, due to the complication of TSP and limitation of the algorithm themselves, they always cannot get better result, such as pr226 and a280 instance. On the other hand, IF-HDPSO algorithm can get the best solution within specified iteration number, which demonstrates IF-HDPSO also has great capability in solving large scale TSP instances.

As the path planning shown in Figure 2, the two cities with shorter distance have greater probability to joint together, the patterning is always raised polygon and not exists intersected node. The proposed IF-HDPSO algorithm, particles get closed to the global optima by the learning factor c continuously, the identical factor function can adjust c so that avoid being trapped in local optima, thus, in the next city selection, it has greater probability to select near city. Intuitionistic fuzzy charisma function allows other particles to participate in the update of velocity and position except the best one, which also can increase the diversity of particle swarm. The elitist reserving strategy preserves admirable individuals in case being destroyed, and replace the low fitness individuals, which can decrease the useless search procedure and the proposed IF-HDPSO needs fewer iterations.

Summary

In this paper, we propose a hybrid method IF-HDPSO, and utilize it to discrete space problem TSP. We apply intuitionistic fuzzy charisma function to represent the corresponding TSP solution, which broadens the application in discrete space. The intuitionistic fuzzy charisma function can select other particles to participate in velocity and position updating. Identical factor function is defined to measure the diversity of swarm and adjust the inertia weight and learning factor. The hybrid of elitist reserving GA and PSO enhance the global search ability and local escape ability. The experimental results indicate IF-HDPSO algorithm is more confidential than other 4 algorithms, and it also can be utilized in discrete spaces.

Acknowledgments

The authors would like to thank the anonymous referees for their careful reading of their helpful comments. This work was supported in part by the National Science Foundation of China under project 61402517, in part by the National Science Foundation of shanxi province under project 2013JQ8035. Hai-tao Mei is the corresponding author. (email: meiht14@163.com)

References

[1] Jinhui Yang, Xiaohu Shi, Maurizio Marchese, et al. An ant colony optimization method for generalized TSP problem[J]. Process in Nature Science, 2008, 18, 1417-1422.
[2] Gilang Kusuma Jati, Suyanto. Evolutionary Discrete Firefly Algorithm for Travelling Salesman Problem[J]. Springer-Verlag Berlin Heidelberg, 2011, 393-403.
[3] Aziz Ouaarab, Xin-She Yang. Discrete cuckoo search algorithm for the travelling salesman problem[J]. Neural Compute & Applic, 2014, 24: 1659-1669.
[4] N. Salmani Niasar, J.Shanbezade, M. M. Perdam, et al. Discrete Fuzzy Particle Swarm Optimization for Solving Traveling Salesman Problem[J]. International Conference on Information and Financial Engineering, 2009, 162-165.
[5] GAN R W, GUO Q S, CHANG H Y, et al. Improved ant colony optimization algorithm for the traveling salesman problems[J]. Journal of Systems Engineering and Electronics, 2010, 21(2): 329-333.
[6] XU Ze-shui. On similarity measures of interval-valued intuitionistic fuzzy sets and their application to pattern recognitions[J]. Journal of Southeast University, 2007, 23(1): 139-143.
[7] TAN Guan-zheng, BAO Kun, Richard Maina Rimiru. A composite particle swarm algorithm for global optimization of multimodal functions[J]. Journal of Central South University, 2010, 21: 1871-1880.
[8] Aziz Ouaarab , Belaïd Ahiod, Xin-She Yang. Discrete cuckoo search algorithm for the travelling salesman problem. Neural Comput & Applic (2014) 24: 1659-1669.