Using a Complex Network Based Optimizer for Solving TSP

Yonghua Zhang, Wenbin Li

Shijiazhuang University of Economics
Shijiazhuang, Hebei, China

***Abstract* — To reveal the influence of the spatial structure of individual interaction on the performance of evolution algorithms, a new algorithm named Complex Network based Optimizer (CNO) is proposed. One individual is placed on a node of a given complex network. Interaction structure of the individuals is determined by the complex network. A sub-population consists of an individual and its neighbors, which achieves evolution with the corresponding evolutionary operators. On four datasets, the performance of CNO is validated. Experimental results show that designing spatial structure is a major weapon for improving the performance of evolutionary algorithms.**

Keywords - complex network; evolutionary algorithm; TSP

I. INTRODUCTION

Spatial structured evolutionary algorithm refers to the evolutionary algorithms where individual interactions are determined by graphics. For example, diffusion model evolutionary algorithm, parallel evolutionary algorithm, celluar automata evolutionary algorithm and distributed evolutionary algorithm all belong to this kind of algorithm [1]. Neighborhood structure (including variable neighbor-hood structure) and space decomposition are two common methods of spatial structured evolutionary algorithm to conduct the spatial structuring. From the local point of view, good neighborhood is the premise to assure the global optimality. Traditional neighborhood structured evolutionary algorithm usually uses one species group. The scale of the group and the problems the group dealing with is small. Space decomposition method divides the group into multiple sub-groups to increase the scale of the group and realize the isolation of the groups. By doing this, the method can increase the diversity of the group during the calculating process and improve the quality of solution. The neighborhood structured method explicitly stated the interactive structure of the individuals during the evolutionary process; although the space decomposition method didn't stipulate the interactive structure, the undecided interactive structure always exists in the whole evolutionary process. According to the research, both of the two methods are effective, which implies: new algorithms can be designed by revealing the relationship between special structure and algorithm performances on the basis of the design of species group structure. However, how to quantify the measurement of "Structure" became the predominant problem. Fortunately, the new complex network theory provided an effective basis, which became an inevitable choice for group special structure[2, 3]. There is reason to believe that new algorithm framework will emerge soon. Evolutionary algorithms based on complex network interaction and noisy multiple targets optimized algorithms will be the hotspots of future researches[3].

This paper designed a complex network based optimizer (CNO). In the CNO, each evolvable individual whose interactive structure is determined by the complex network was placed at each node of the complex network. A small species group will be formed by these individuals with their neighbors and evolved by relevant evolutionary algorithms. On the one hand, the species groups exchange information through the network structure; on the other hand, they exchange the information by the algorithm strategy. In this paper, the interactive structure of the individuals is static.

II. RESEARCH MOTIVES AND SCALE-FREE NETWORK MODELING ALGORITHM

A. *Research Motives*

When talking about evolution or group intellectual algorithm, individual and interaction will be always put forward. To some extent, the evolutionary algorithm is solved through the realization of individual interaction. For example, the intersection of genetic algorithm can be regarded as the interaction of father and mother; the solution of PSO algorithm relies on the interaction between the current individual and the optimal individual (or neighboring individual). Ants in ant colony algorithm interact through pheromone. Obviously, in traditional evolutionary algorithm, the interaction structure of individuals is dynamic and it is hard to record the structure. According to the principle of "Structure determines function", it is believed that good static interaction structure design can help the algorithm get better performance. Actually, this idea is not the first one. K.M. Bryden et al. proposed an evolutionary algorithm based on graphic. In this algorithm, the author used combing graphics to restrict the choosing of individual interaction partners[4]. The combing graphics also restricted the speed and way of information transmission to make the competitive solutions more "mature" in sufficient time. Zhuang Jian et al. re-designed and improved the evolutionary algorithm on the basis of complex system and complex network idea[5]. They applied the self-organizing and self-adaptive features from complex system theory to the design of genetic algorithm, and reconsidered the equilibrium relationship between the structure and parameters of the genetic algorithm. This model has a very good effect in the optimization of high dimension functions. The author and the team of this paper also attempted to solve TSP problems with the combination of complex network and evolutionary algorithm, and also made some progress [6].

B. *Scale-free Network Modeling Algorithm*

Considering the fact that most of the degree distributions of real network are power-law distribution (meaning most of real network are scale-free network), this paper restricted the special structure of individual interaction in a pre-generated scale-free network. The degree of scale-free network is seriously asymmetrically distributed: few nodes have many connections, which play a dominant role in the operation of scale-free network; most of the nodes have very few connections. Algorithm 1 refers to the pseudocode of BA model realized by this paper[7].

Algorithm 1 Pseudocode of BA model realized by this paper.

Input: Given the network with m0 nodes; given the evolutionary steps t.

Output: scale-free network

Process:

FOR (i=1;i<=t;++i) //totally t steps were experienced

BEGIN

//Prior to nodes with larger connection

Arrange with descending order according to node degree; order results were recorded in nodes;

All the nodes in the network were marked with unselected;

Increase a new node in the network; //increase node

FOR (j=0;j<=m // all the nodes were selected;)

BEGIN

Select the first unselected node from nodes, denoted as node.

//D referring the sum of degree of all the nodes

Produce a randomrefers to the edge set. The mathematical model of TSP is shown as following.

Each possible solution can be expressed as a vertex sequence $V_1, V_2, \cdots, V_N, V_1$, where x,y=1, …, N, $V_x \neq V_y, V_x, V_y \in V$, same as below. Solution space is a sequence set of all possible vertex sequences, which makes the following formula with the least value:

$$D(v_1, v_2, \ldots, v_N, v_1) = d(v_N, v_1) + \sum_{x=1}^{N} d(v_x, v_{x+1}) \tag{1}$$

Where $d(v_x, v_y)$ stands for the weight on edge (v_x, v_y) (distance between city v_x and city v_y.)

In the typical TSP problem, $d(v_x, v_y) = d(v_y, v_x)$. Hence, typical TSP problem is also called the symmetrical TSP.

C. *Design of Complex Network Optimizer for the Solution of TSP Problem*

The solving process of CNO algorithm depends on the complex network constructed in advance. Each node in the complex network represents a solution (individual). The encoding method of each individual can be any of the binary, integer-coded, floating-point and mixed methods[8]. A small species group will be formed by a node and its neighboring nodes. The evolution of the small species group is actually the evolution of the individuals in the group. The evolutionary ways of the individual include mutation method and hybrid method (same as the genetic algorithm). The evolution of individual may be influenced only by the small species group or by both of small species group or big species group. After several times of iterations, all the species groups in the complex network would stop evolution (as shown in Figure 1).

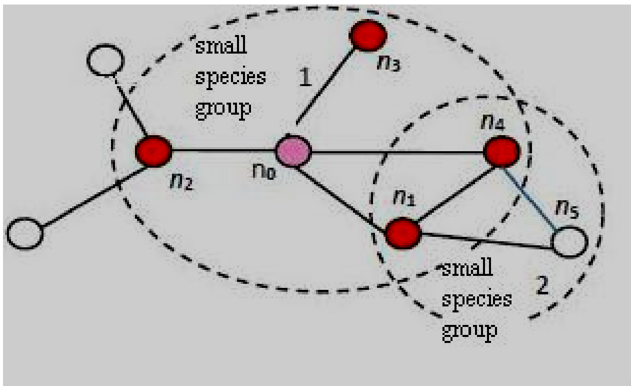


Figure 1. Diagram of Small Species Group

In allusion to TSP problem, Algorithm 2 expressed each node code as N+1 (N refers to the number of cites, same as below) dimensional vectors: $< C_1, C_2, \cdots, C_N, C_1 >$, where each vector $c_i (i \in c_i \{1, 2 \cdots N\})$ uses real number encoding method to represent the city number. In the first N dimensions, each vector is unequal. The vector of N+1 dimension is the same as that of the first dimension. The physical meaning shows the solution: from c1 city to c2,… to cN city and back to c1 city. The calculating formula of the individual fitness can be defined as Formula (2):

$$F(c_1, c_2, \ldots, c_N, c_1) = 1 / (d(c_N, c_1) + \sum_{x=1}^{N-1} d(c_x, c_{x+1})) \tag{2}$$

Where, $d(c_x, c_y)$ refers to the weight on the edge (c_x, c_y) (distance from city cx to city cy), $x \neq y, x, y \in \{1, 2 \cdots N\}$.

Algorithm 2. CNO algorithm

Input: Complex network, TSP problem

Output: Optimal solution

Process:

Initialize the code of each node in the network;

Calculate the fitness of each node;

DO

For each node

Find out all the neighboring nodes of current node (noted as small species group);

Evolution of the small species group (including current node);

END

WHILE (terminate condition): //Iteration times is the terminate term of this paper;

Output the node with best fitness;

The evolutionary strategy of small species group used in this experiment is shown as Algorithm 3. Algorithm 3 only used mutation operator rather than crossover strategy. There are three types of mutation methods. Taking $< C_1, C_2, \cdots, C_N, C_1 >$ as the individual code, and $I, J(I, J \in \{1, 2, \cdots N\})$ as the mutation node, the first mutation method is to turn over the vectors between I dimension and J dimension and get $< C_1 \cdots C_{I-1}, C_J C_{J-1}, \cdots, C_{I+1}, C_1, C_{J+!}, C_N, C_1 >$. The second mutation method is to exchange the vectors on I dimension and J dimension and get $< C_{I-1}, C_J C_{J-1}, \cdots, C_{I+1}, C_1, C_{J+!}, C_N, C_1 >$. The third mutation method is to slide $C_{I+1}, \cdots, C_{J-!}, C_J >$ to the front of cI and get $< C_1 \cdots C_{I-1}, C_{I+1}, \cdots, C_{J-1}, C_J, C_{J+1}, C_N, C_1 >$.

Algorithm 3. Evolution of small species group

Input: Small species group, original species group

Output: Evolved small species group

Process:

IF the individual number of small species group is not the multiple of 4

Extract several individuals from the original species group to the small species group to make the individual number the multiple of 4;

ENDIF

Randomly arrange the individuals in the small species group;

FOR (p=4; p＜＝individual number of small species group; p+=4)

fp=4 individuals with subscripts from p-3 to p extracted from the small species group;

The optimal individual from $p_b = fp$;

Mutate pb with three methods respectively and get three individuals of p1, p2 and p3;

Replace the other three individuals respectively in fp except pb with p1, p2 and p3.

ENDFOR

Return to the evolved small species group;

From the view of crossover strategy, CNO is almost the same as traditional genetic algorithm. However, there are significant differences between the two algorithms. First, based on the given complex network, CNO created multiple structured information exchange space (or small species groups) for the individuals. As shown in Figure 1, node n1 and node n4 coexists in the “small species group 1” and “small species group 2”. They can evolve in any of the two groups. Besides, CNO provides the implied choice of “survival of the fittest” instead of explicit selection operator. As shown from Algorithm 3, CNO reserved superior individuals for the small species group. Inferior individuals were replaced by superior mutant individuals (inferior individuals will be eliminated). However, it’s worth nothing that CNO didn’t require the replacer must be superior compared with the individual to be eliminated. Obviously, selection probability is not constant in CNO. The implicit existence of selection operator makes the CNO algorithm convergent. In addition, one individual only has one chance of evolution in iteration in the traditional genetic algorithm. However, the individual has multiple chances of evolution in the iteration of CNO. According to figure 1, in one certain iteration, n1 and n4 may evolve either when n0 or n5 is the current individual.

In algorithm 3, there are two information exchange ways between small spices groups: 1). When the individual number is not the multiple of 4, several individuals from other groups will be introduced to this species group. With these individuals, information exchange will be realized between the small species groups. 2). Realize the information exchange by the individuals crossing small species groups. For example, in Figure 1, “small species group 1” and “small species group 2” use n1 and n4 to exchange their information.

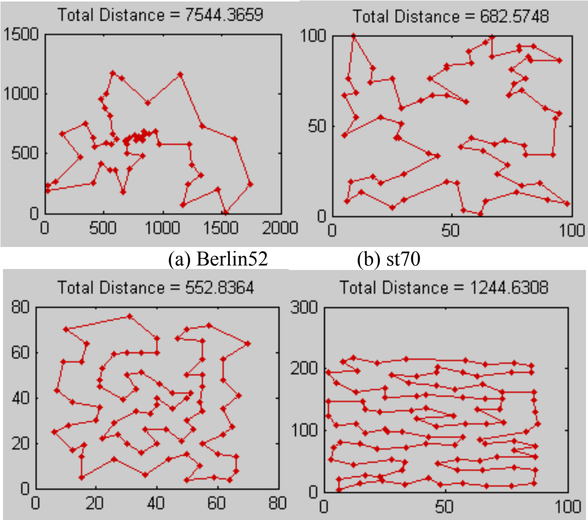
IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. *Experiment Design*

This paper conducted a series of simulation experiment. Hardware platform: DELL XPS Ⅰ 14Z notebook. Software: Windows 7 + Matlab R2012a. The 4 data sets adopted in the experiment were provided by TSPLIB. The complex network used in the experiment is a 300 nodes scale-free network generated by Algorithm 1. The operating parameters of CNO are: 300 species group scale and 40 iteration times. Obviously, the operating parameters set for CNO algorithm are very limited and it is easy to select the parameter value. Each experiment will be operated 10 times independently and recorded separately: 1) the best result among the 10 experiments, showing the optimizing ability of CNO; 2) the optimal solution among the 10 experiments, showing the stability of CNO performance.

B. *Experimental Results and Analysis*

The optimal solutions of CNO based on the 4 data sets are shown respectively in Figure 2(a)-(d). As shown in Figure 2, on the 4 data sets (shown as Table 1), the optimal solutions that CNO obtained are 7544, 682, 552 and 1244, which are close to international optimal solutions. Especially on the berlin52 data set, the difference with international optimal solution is only 2. According to the experimental results, the more the cities, the greater the difference between CNO and international optimal solution will be. Future study will focus on the reason of the problem and improve CNO algorithm according to the large scale of TSP problems. As observed in the experiment, during the initial stage of iteration, the convergence rate of CNO is very fast. But there is no premature convergence in this period. Specifically, after 232 times of iteration, CNO finally found the optimal solution of 7544 for berlin52; the optimal solution of 682 for st70 in the 290 iteration; the optimal solution of 552 for eli70 in the 365 iteration; the optimal solution of 1244 for rat99 in the 373 iteration. Due to the limited space of the article, here left out the diagram of CNO convergence results(as shown in Figure 2).



(a) Berlin52 (b) st70

(c) eli76 (d) rat99

Figure 2. The Optimal Solutions to the TSP Problems Found by CNO

Table 1 has presented the optimal solutions that CNO found for the data sets during the 10 independent operations. The results showed that, the solution of 7544 was found for many times in berlin52. In st70 and eli7, the solutions are close. In rat99, some inferior solutions may exist. Hence, CNO needs to be further improved regarding the large scale TSP problems.

TABLE 2 OPTIMAL SOLUTIONS OBTAINED IN THE 10 TIMES OF INDEPENDENT OPERATIONS (THE BOLD ITEMS REFER TO THE OPTIMAL SOLUTIONS IN EACH COLUMN)

berlin52	st70	eli76	rat99
7.7469e+03	688.6961	552.8364	1.2826e+03
8.0142e+03	687.9468	553.8066	1.2850e+03
7.5444e+03	695.6141	559.9217	1.2700e+03
7.5444e+03	694.4258	582.2496	1.2519e+03
7.9795e+03	682.6627	558.7959	1.2662e+03
8.0905e+03	682.5748	563.0288	1.2717e+03
7.7167e+03	701.5511	559.0301	1.2590e+03
7.5444e+03	695.7006	569.6481	1.3175e+03
7.5444e+03	686.4978	558.2592	1.3029e+03
7.5444e+03	700.8438	562.0959	12446e+03

V. CONCLUSION

J. L. Payne et al. pointed out that the species group structure of evolutionary algorithm decided the transmitting and mixing of superior alleles and would affect the searching performance[9]. The network structure existed in the interactive individuals is considered as one of the key factors of influencing the emerging results of individual cooperative behaviors. With the development of complex network research, the combination of complex system or complex network with evolutionary computation will be an inevitable trend.

This paper attempted to combine the complex network with evolutionary algorithm and designed a complex network optimizer (CNO). Based on given network, CNO uses the complex network as the spatial structure for the individual interactions and defined the small species group and the evolutionary methods of them. On the 4 TSP problems, CNO performed well and proved the rationality of its design. Future work will consider the influences of dynamic network structure on the algorithm performances and try to improve and upgrade the solutions to large-scale TSP problems.

ACKNOWLEDGMENT

Sports science and technology research of Project in Hebei Province, Research on the construction of sports talent database based on big data(Project No. 20153006).

REFERENCES

[1] A. E. Eiben, J. E. Smith. "Introduction to evolutionary computing", Chapter Multi-modal problems and spatial distribution. Springer Verlag, NO. 9, pp.153-172, 2007.

[2] A. Gasparri, S. Panzieri, F. Pascucci, and G. Ulivi. "A spatially structured genetic algorithm over complex networks for mobile robot localization", IEEE Int. Conf. Robot. Autom, Piscataway, pp. 4277-4282, 2007.

[3] A. M. Zhou, B. Y. Qu, H. Li and S. Z. Zhao, et al. "Multiobjective evolutionary algorithms: a survey of the state of the art", Swarm and Evolutionary Computation, vol. 1, No. 01, pp. 32-49, 2011.

[4] K. M. Bryden, D. A. Ashlock, S. Corns, et al.. "Graph-based evolutionary algorithms", IEEE Transactions on Evolutionary Computation, vol. 10, No. 05, pp. 550-567, 2006.

[5] Jian Zhuang, Qingyu Yang, Haifeng Du. "High Efficient Complex System Genetic Algorithm", Journal of Software, vol. 21, No. 11, pp. 2790-2801, 2010.

[6] Yiying Chen, Zexing Zhang, Wenbin Li. "An Algorithm Constrained by Complex Network's Structure for Solving TSP", ISDEA, vol. 10, No. 02, pp. 270-273, 2013.

[7] A. L. Barabasi, R. Albert. "Emergence of scaling in random networks", Science, vol. 17, No. 09, pp. 509-512, 1999.

[8] Yongjian Chen, Yan Zhou. "Application of Coding Genetic Algorithm and its Improvement Research", Journal of Taiyuan Normal University(Natural Science Edition), vol. 7, No. 02, pp. 76-77+95, 2008.

[9] J. L. Payne, M. Giacobini, J. H. Moore. "Complex and dynamic population structures: synthesis, open questions, and future directions", Soft Comput, vol. 17, No. 07, pp. 1109-1120, 2013.