Solving TSP with Novel Local Search Heuristic Genetic Algorithms

Jianxin Zhang
Information Engineering School, Beijing University of Science and Technology, Beijing 100083, China
College of Information Engineering，Inner Mongolia University of Technology，Hohhot 010062, Inner Mongolia，China
b2005zjx@163.com

Chaonan Tong
Information Engineering School, Beijing University of Science and Technology, Beijing 100083, China
tcn@ies.ustb.edu.cn

Abstract

The standard genetic algorithm often meets the occurrence of slow convergence for solving combinatorial optimization. In this study, we present a hybrid genetic algorithm (LSHGA) for symmetric traveling salesman problem. In the algorithm, a modified local search method based on new neighborhood model was contrived as crossover operation. And a MUT3 operator was introduced as mutation operation. For evading plunge into local extremum, some new ideas are incorporated in our algorithm. The strategy that unites stochastic tournament and elite reservation was used in the process of selection. An idea of reservation ratio was put forward, and the theory of self-adaptive was employed for conforming parameters of LSHGA at the same time. As a result, the population diversity of LSHGA was preserved, and can accomplish convergence quickly. Eight example problems in TSPLIB were showed in the paper to demonstrate the capabilities of the proposed algorithm.

1. Introduction

Traveling Salesman Problem (TSP) is classical combinatorial optimization problem. Some combinatorial optimization problems are NP-hard problems, TSP is one of them. Effective method for solving a NP-hard problem is that utilize some heuristic algorithms for obtaining approximate result. This paper presents a modified genetic algorithm (LSHGA), which combines Local Search and Genetic Algorithm, for Symmetric TSP

Local Search algorithms are effective heuristics method for many Combinatorial Optimization problems. However there are a lot of issues where the size of the neighborhood increases exponentially as the size of issue increases. It is time-consuming for getting famous solution. The choice of the neighborhood structure and the design of the algorithm are a critical issue for contriving a good heuristic algorithm. The neighborhood structure adapt to TSP is k-Opt Shift. The k-Opt Shift is foundation for three distinguished Local heuristics Search: 2-Opt, 3-Opt and Lin-Kernighan (LK). They determine the neighborhood structure of TSP [1]. And many articles contributed the improvement of LK [2, 3]. However, the critical disadvantage of Local Search is easily to run into local minimum. Combining with other heuristic algorithm, such as Genetic Algorithm, is a good idea for conquering the problem.

Simulated annealing and Tabu search are effectual means for TSP. Meer [4] improved a simulated annealing algorithm based on Metropolis algorithm. Utilizing the concept of genetic algorithm, Ning [5] contrived a Tabu search algorithm which hold crossover operator to solve TSP.

Genetic Algorithms which act on the basis of evaluation and selection mechanism in nature, belongs to an adaptive stochastic search algorithms. Since simple structure and implicit parallelism, GA is powerful tools used widely in Combinatorial Optimization problems. The design of code, crossover operator and mutation operator are key issue for TSP. Many researches commit themselves to improve GA or combine GA with other methods [6-9].

2. Modified Local Search Heuristics Genetic Algorithm (LSHGA)

The LSHGA which presents in this paper for Symmetric TSP is based on GA frame and asexual reproduction [8]. In the algorithm, a modified local

978-0-7695-3304-9/08 $25.00 © 2008 IEEE
DOI 10.1109/ICNC.2008.235

search method based on new neighborhood model was contrived in crossover operation. And a new MUT3 operator was introduced as mutation operation. Since employing local search, the critical problem of designing the algorithm is to preventing get into local extreme and preserving population diversity. So some efficacious mechanisms are introduced into LSHGA. The strategy that unites stochastic tournament and elite reservation was used in the process of selection. An idea of reservation ratio was put forward, and the theory of self-adaptive was employed for conforming parameters of LSHGA.

2.1. Encoding and Fitness Function

For solving the TSP by GA, there are two kinds encoding manner: natural encoding of a tour and Grefenstence. The natural encoding of a tour is the most natural representation of a TSP tour, as an ordered list of the cities to be visited. Merits of the natural encoding manner are explicitness and concision, and the process of decode is not complex. However, if make use of standard crossover operator, the crossover operation produces illegal paths. To avoid the problem posed by permutation representation, some amendatory crossover operator were brought forward: PMX、CX、OX [9]. But these crossover operations have obscure encoding. All other code manners except the natural encoding of a tour require complicated decodes approach, which consumes large numbers of computer time; especially solve large-scale TSP. To keep away from the intricate process of decode, the natural encoding of a tour is used in the article.

The fitness of a solution is inversely proportion to the length or cost of the solution. i.e., if the length of a tour is greater, the fitness of the tour is less. Equation (1) is introduced for calculating the fitness. The number of cities will be denoted by *Ncities*. The Fitness Function of LSHGA is:

$$f(T) = Ncities \,/\, \Gamma(T) \qquad (1)$$

where Γ(T) is the length of tour T.

2.2. Design of Crossover Operator

To avoid infeasible solution, the scheme of asexual reproduction is betaken, i.e., one parent is chosen into crossover operation and one child is gained. The crossover operator is a modified local search approach whose neighborhood structure is achieved by executing improved Lin-Kernighan. To be more specific, the modified local search is executed on the chosen parent individual, which is the benchmark of the iteration, for achieving a better fitness solution in neighborhood structure. The neighborhood structure, that is, the manner in which the neighborhood is defined, is illustrated in Figure 1. The first solution which is better than the parent is found in the neighborhood as the child of crossover operation. The progress of improved Local Search algorithm is as follows: assume that the parent chromosome represent the tour T.

Step 1: Randomly choose two nodes in parent tour T, they are denoted by i and j respectively. Deletes the arc (i, j) which links the two nodes, creating the Hamiltonian path;

Step 2: One of the end points of the Hamiltonian path is fixed (assume node i is fixed) and stays fixed until the end of the iteration. Choose a node k whose distance to node j is least than other nodes. Add the arc (k, j), giving the stem and cycle [9];

Step 3: Delete the arc (k-1, k) and the arc (k, k+1), add the arc (k, j), having a new Hamiltonian path, i.e., a new tour T';

Step 4: Calculate the fitness of tour T'. If the fitness of tour T' is better than the one of tour T, then T' is the child chromosome of T, and the iteration is over, else go to Step5;

Step 5: If the child chromosome is not discovered as exhausting all nodes in the tour T, then the parent tour is regard as the child tour, and the iteration is over, else go to Step1.

The crossover operation for TSP with 7 nodes is simply illustrated in Figure 1.

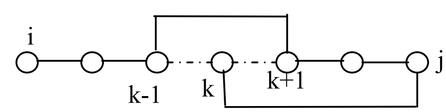

Figure 1. Illustrating modified neighborhood structure: a tour on 7 nodes

The novel crossover method can drive the execution of genetic operation to enhancing the fitness of chromosome, and can accelerate convergence of the solution.

2.3. Design of Mutation Operator

A MUT3 operator is employed as the mutation operator in the study [8]. Its process is described as following: Three position numbers are chosen randomly in a chromosome. The chromosome is broken at these positions, so it is divided into 3 parts. The tour pieces are rearranged to create a child of this mutation. For example, suppose a mutation takes place for solution *T*, denoted by *ABC*. The lengths of gene pieces are *p1*, *p2* and *p3*, respectively. Let *C'* is the inverse permutation of *C*, i.e.

$$C' = c_{p3} c_{p3-1} \cdots c_1$$

The individual *ABC* is rearranged in term of the mode of *BC'A*, the mutated solution *T'* is given by *T'*:

$$T' = b_1 b_2 \cdots b_{p2} c_{p3} c_{p3-1} \cdots c_1 a_1 a_2 \cdots a_{p1}$$

After mutation course, there is remarkable difference between patent and child solution on genotype (phenotype), so the population diversity is enhanced. Further more, the child individual which is obtained by the MUT3 operator is legal tour.

2.4. Recalculation of Fitness

According to the Equation (1), the great task is to calculate the length of tour. The length relation of parent tour to child tour is given in Equation (2):

$$\Gamma(T') = \Gamma(T) + \Delta d \qquad (2)$$

where Γ(T') and Γ(T) is the length of child tour and parent tour respectively; *Δd* is the difference of the length.

So for computing the length of every child individual of crossover or mutation, only *Δd* is calculated:

If the child tour comes from crossover operation:

$$\Delta d = [d(i,k) + d(k-1,k+1) + d(k,j)] - [d(i,j) + d(k-1,k) + d(k,k+1)]$$

Otherwise:

$$\Delta d = [d(b_{p2}, c_{p3}) + d(c_1, a_1) + d(a_{p1}, b_1) - [d(a_{p1}, b_1) + d(b_{p2}, c_1) + d(c_{p3}, a_1)]$$

where *d (m, n)* is the distance of node m and n.

2.5. Establish breeding pool and the selection scheme

All parent individuals and children individuals are put into the breeding pool as candidate solution for selection. The size of breeding pool is double the size of initial population. If the number of chromosome in current breeding pool is less than the size of breeding pool, some fresh individuals who are created irregularly will be introduced. The mechanism effectively ameliorates the population diversity and restricts the premature convergence of the population.

Traditional simple genetic algorithms select the better individuals from the population into the next generation based on the roulette wheel selection. This result is that super-individual will take over most of the population in a few generations. LSHGA is combined with the binary stochastic tournament selection so that the population diversity is kept well all along.

The elitism, that can improve the performance of a genetic algorithm, is employed, i.e., the m best solution as elite will be copy to the next generation. In this study, we present a new concept: reservation ratio.

Definition 1: The reservation ratio is the ratio of the number of elite (m) and the size of breeding pool, and is denoted by P_e, as a parameter of LSHGA:

$$P_e = \frac{m}{Pop} \qquad (3)$$

where *Pop* is the size of breeding pool.

The value of m is confirmed by Equation (3). Moreover, the P_e is the function of the variance of the fitness of whole population, which is described in Equation (6).

2.6. Crossover rate, Mutation rate and Reservation ratio

The Crossover rate Pc can be gained from the Equation (4), which indicates that the Crossover rate is self-adaptive with the fitness of population.

$$P_c = \begin{cases} k_c(f_{\max} - f)/(f_{\max} - \bar{f}) & f > \bar{f} \\ k_c & f < \bar{f} \end{cases} \qquad (4)$$

where f_{max} is the maximal fitness in currently generation; $\bar{f}$ denote average fitness; *f* is the fitness of the parent individual of crossover operation; k_c is constant, here k_c=1.0.

According to the Equation (4), the solution with less fitness has greater chance to executing local search for finding a better child solution. Whereas the individual possess better fitness get the chance with less probability. Therefore the difference of fitness among population is decreased, and the situation getting into relative extremum may be avoided. The advantages of the policy are keeping the population diversity and preventing premature convergence.

After the crossover progress, if the better solution is not found, the parent should be changed by the mutation operation with mutation rate. In LSHGA, the mutation rate is a function of the population diversity. A conception is introduced as a measure of the population diversity.

Definition 2: Suppose that *S* is a tour of TSP, that is, it is a set of edges. Let |*S*| denote its cardinality and *T* is other tour. Then *S-T* is the set of edges that appear in *S* but not in *T*. We define the *distance D(S, T)* =|*S-T*|+|*T-S*|, i.e., the number of edges that appear in *S* or *T* but not both.

The *distance D(S, T)* is a measure of the different among individual. The upper bound of *D(S, T)* is 2*N*, *N* is the number of nodes in TSP. The population diversity is reflected by all distance in the currently generation population.

When the solution got into local optima, it should be mutated with greater mutation probability for getting away from the local optima. However in the latter pare of the algorithm, the mutation rate should be decreased for accelerating the convergence of the algorithm. The Mutation rate P_m is confirmed by the Equation (5):

$$P_m = \begin{cases} k_m \dfrac{D^t_{\min}(S,T)}{2N} & f = f' \\ 0 & f > f' \end{cases} \quad (5)$$

where, S denote the child individual which come from mutation operation; T is the tour whose distance to S is minimum compared with all other nodes, $S \neq T$, and $T \in J(t)$, $J(t)$ is the set all chromosome in the t^{th} generation; denotes the minimal distance from T to S; k_m is constant, here k_m=0.5; f is the fitness of S, f' is the fitness of the parent of S.

After mutation, if the fitness of child chromosome S is not improved, and the difference of S and the other individual is relative big value, the solution S may have run into local extreme. Therefore, according as Equation (5), the P_m becomes a greater number for making S to escape from local optima. While the algorithm is near of the end phase, the difference among population trail off, i.e., the value of $D^t_{\min}(S,T)$ reduces. So the P_m decrease too, and making the algorithm convergent quickly.

The reservation ratio P_e in Equation (3) is gained from the Equation (6) below:

$$P_e = k_e \frac{\overline{f^2} - \overline{f}^2}{f^2_{\max} - \overline{f}^2} \quad (6)$$

where k_e is constant, here k_e=0.1; the signification of other parameter is same as in Equation (4).

According to the Equation (6), while the variance of the fitness of whole population is greater value, P_e is also relative great value, i.e., more individual whose fitness is better is survival into the next generation. The convergence speed of LSHGA is enhanced. On the other hand, while the variance of the fitness of whole population is smaller value, P_e becomes a lesser value, preserving population diversity, evading plunge into local minimum.

Table. 2 Comparison between LSHGA and GGA based on symmetric TSP

	LSHGA			GGA		
	PS	CN	BS	PS	CN	BS
eil51	200	300	426	150	2000	428.98
eil76	260	300	538	200	2000	553.70
eil101	300	400	640	105	5000	665.50

2.7. Initial population

Initial population is yielded randomly accord with uniform distribution in LSHGA. So these individuals scatter the solution space, ensuring population diversity, avoiding getting into relative extreme.

3 Experiments and Results

In this section, we present the experimental results on eight symmetric traveling salesman problems in TSPLIB [10] and compare LSHGA to GGA [11] with 3 TSPs, which belong to the eight problems. We implement the LSHGA on 1.7GHz Pentium M CPU running Matlab 7.0. LSHGA is performed 20 independent runs respectively to all the eight problems.

For evaluating the quality of solution with LSHGA, relative error is made use of and denoted by γ:

$$\gamma = \frac{C - C^*}{C^*} \times 100\ \% \quad (7)$$

where C is the solution of LSHGA; C^* is the solution in [10]. The solutions of 8 TSP instance with LSHGA and the parameters of the algorithm are listed in table 1. Moreover the experimental result compared with GGA is showed in table 2. In the table 1 and table 2, notations, OS, BS, PS and CN, denotes the optimal result in the TSPLIB, the best solution of the approach, the population size of the algorithm and the convergence iteration number respectively.

While the scale of problem increases, the dimension of feasible solution space extends exponentially. However, as long as the population size of LSHGA is augmented reasonably, the satisfied result will be gained, and the convergent iteration number only increases finitely. For the eight problems in table 1, all relative error is less than 1.7%. The result demonstrates clearly that LSHGA is effective for solving some TSP. In table 2, it is obviously that the solutions and the convergence number of LSHGA are better apparently than the ones of the GGA for problem eil51, eil76 and eil101.

Table 1. The result of solution with LSHGA

	PS	CN	OS	BS	γ
eil51	200	300	426	426	0
eil76	260	300	538	538	0
eil101	300	400	629	640	1.75%
ch130	350	500	6110	6164	0.88%
ch150	400	750	6528	6606	1.19%
kroA100	300	400	21282	21296	0.66%
kroA150	450	800	26524	26775	0.95%
kroA200	500	1100	29368	29843	1.62%

In figure 2 to figure 4, the best solution tours of the LSHGA among 20 independent executions are illustrated for TSP instance ch130, ch150 and kroA100 respectively.

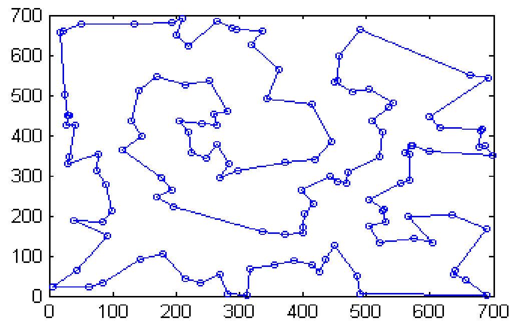

Figure 2. The best tour of ch130 with LSHGA

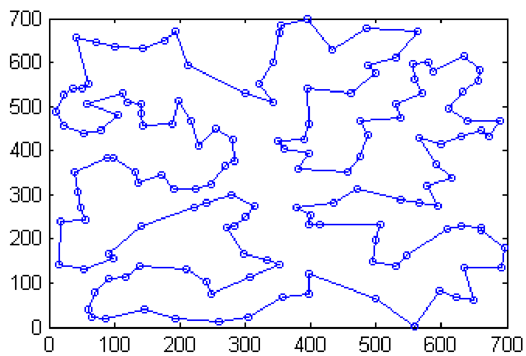

Figure 3. The best tour of ch150 with LSHGA

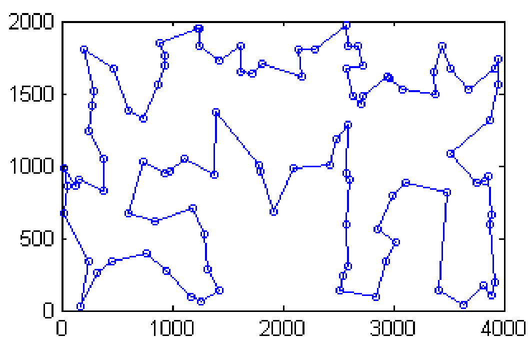

Figure 4. The best tour of kroA100 with LSHGA

4. Conclusion and Suggestions

In this study, we design a novel neighborhood structure for TSP, and the Local Search approach combine with GA to form the LSHGA algorithm. The LSHGA algorithm utilizes the properties of many individual co evolutionary and parallel search of GA, and conquers the defect of local search, which is easily to get into relative optimum. Some effectual ideal are proposed for preserving the population diversity, preventing precocity, and enhancing the speed of convergence. By and large, the algorithm can achieve satisfied solutions, and accomplish convergence quickly for middling-scale TSP. However, the run speed is comparatively slow in the case of large-size TSP.

5. References

[1] Ahuja R K, Ergun Ö, Orlin J B. A survey of very large-scale neighborhood search techniques. Discrete Applied Mathematics, 2002, 123(1-3):75-102

[2] Gamboa D, Rego C, Glover F. Data structures and ejection chains for solving large-scale traveling salesman problems. European Journal of Operational Research, 2005, 160(1):154-171

[3] Applegate D, Cook W, Rohe A. Chained Lin-Kernighan for large traveling salesman problems. INFORMS Journal on Computing, 2003, 15(1):82-92

[4] Meer K. Simulated annealing versus Metropolis for a TSP instance. Information Processing Letters, 2007,104(6):216-219

[5] Ning Yang, Ping Li, Baisha Mei. An angle-based crossover tabu search for the traveling salesman problem. In: Proceedings of the Third International Conference on Natural Computation, Haikou, China,2007,4:512-516

[6] Peng Gang, Nakatsuru Takeshi. Efficiency of local genetic algorithm in parallel processing. In: Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies, Dalian, China, 2005:620- 623

[7] Katayama K, Sakamoto H. The Efficiency of Hybrid Mutation Genetic Algorithm for the Travelling Salesman Problem. Mathematical and Computer Modelling. 2000,30:197-203

[8] Chatterjee S, Carrera C, Lynch L A. Genetic algorithms and traveling salesman problems. European Journal of Operational Reserch, 1996,93(3):490-510

[9] Takahashi R. Solving the traveling salesman problem through genetic algorithms with changing crossover operators. In: Proceedings of the Fourth International Conference on Machine Learning and Applications, Los Angeles, CA, USA ,2005:319-324

[10]http://www.iwr.uni-heidelberg.de/iwr/comopt/soft/TSPLIB95 / TSPLIB.html

[11]Yingzi Wei, Yulan Hu, Kanfeng Gu. Parallel search strategies for TSPs using a greedy genetic algorithm. In: Proceedings of the Third International Conference on Natural Computation, Haikou, China,2007, 3:786-790