MATEC Web of Conferences **100**, 02025 (2017) DOI: 10.1051/matecconf/201710002025
GCMM 2016

Optimization of Multiple Traveling Salesman Problem Based on Simulated Annealing Genetic Algorithm

Mingji Xu[1,*], Sheng Li[1] and Jian Guo[1]

[1]School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China
Corresponding Email: jahsonxu@hotmail.com

Abstract. It is very effective to solve the multi variable optimization problem by using hierarchical genetic algorithm. This thesis analyzes both advantages and disadvantages of hierarchical genetic algorithm and puts forward an improved simulated annealing genetic algorithm. The new algorithm is applied to solve the multiple traveling salesman problem, which can improve the performance of the solution. First, it improves the design of chromosomes hierarchical structure in terms of redundant hierarchical algorithm, and it suggests a suffix design of chromosomes; Second, concerning to some premature problems of genetic algorithm, it proposes a self-identify crossover operator and mutation; Third, when it comes to the problem of weak ability of local search of genetic algorithm, it stretches the fitness by mixing genetic algorithm with simulated annealing algorithm. Forth, it emulates the problems of N traveling salesmen and M cities so as to verify its feasibility. The simulation and calculation shows that this improved algorithm can be quickly converged to a best global solution, which means the algorithm is encouraging in practical uses.

1 Introduction

In recent years, there are many multiple Traveling Salesman Problems [1] in reality as society develops forward. For example, problems of multiple watering cart routing [2], urban medium-voltage distribution network planning [3], ventilation network, aircraft assembly sequence planning [4], hot rolling orders staffing issues in steel industry [5], etc.

Multiple Traveling Salesman Problem is evolved from Traveling Salesman Problem. This issue is involved in allocation between several tasks in several individuals [6]. Therefore, compared to pure path optimization in Traveling Salesman Problem, multiple Traveling Salesman Problem seems to be more complicated and challenging due to task allocation and dispatch.

From the points of MTSP description types, it can be divided into four cases [7]: M traveling salesmen set off at the same city and back to that city together; M traveling salesmen set off at N different cities and back to their own leaving city; M traveling salesmen set off at the same city but back to N different cities; M traveling salesmen set off at N different cities but back to the same city. While this paper studies and simulates the first description type.

Looking from MTSP solving ways, it can be divided into following four ways; MTSP's degradation to TSP; exact algorithm; heuristic algorithm and Meta-heuristic algorithm. The research on MTSP has a history of fifty years. At first exact algorithm was adopted to solve the problem, which used a lot Branch-and-bound algorithm [8]. While as Traveling Salesman increases and city enriches, it would spend more time on exact algorithm so it is replaced by heuristic algorithm. The representative of heuristic algorithm Lin-kernighan, but it would make space more complicated if MSTP is degraded to TSP. Genetic algorithm, ant colony algorithm, neural network are typical of Meta-heuristic algorithm. In recent decade, researches on Hierarchical Genetic Algorithm increases a lot. Different chromosome codings result in different solution space and it will have an impact on convergence and optimization of algorithm [9]. Compared to ordinary chromosome, hierarchical chromosome has a stronger binding capacity towards to information and their gene cluster can control to each other. Therefore, this kind of algorithm could solve multi-variables optimization problems that normal genetic algorithm can't deal with, but in terms of large dealing space and redundant solution, it will remain to be solved later [10].

This paper improves the chromosome coding of hierarchical structure, reduces solution space; designs self-adaptive genetic operator for crossover and mutation; modifies Simulated Annealing Algorithm and makes it apply in MTSP solution with hierarchical genetic algorithm.

2 Improved simulated annealing genetic algorithm

Necessary coding on chromosome structure can reduce solution space and enhance operation efficiency of algorithm [11]. The design of self-adaptive genetic operator for crossover and mutation could avoid premature convergence problem efficiently and protect population diversity. Improve simulated annealing genetic algorithm, blend it with genetic algorithm and stretch it in fitness can make algorithm get rid of partial optimization trap.

2.1 Suffix coding design of chromosome

The 'Node Accession' coding design of chromosome, like Figure 1, which could make multiple Traveling Salesman Problems degrade to single Traveling Salesman Problems [12]. Take the example of MTSP: m Traveling Salesman, n cities, this design may produce (n+m+1)! solutions, and other abundant solutions.

Hierarchical coding on chromosome, like Figure2, the upper parameter gene cluster is controlled by the bottom controlling gene cluster. Compared to ordinary chromosome, hierarchical chromosome has a stronger binding capacity which is beneficial to multi-parameter optimization problems. Also, taking the example of MTSP: m Traveling Salesman, n cities, there will be $n!m^n$ probable solutions and other abundant solutions [13].

The improvement of hierarchical coding on chromosome changes bottom controlling gene cluster into city quantities of each traveling salesman passed and it suffixes the upper parameter gene cluster, just like what is shown in Figure 3. This design sets the visiting sequence of traveling salesman [14], thus producing $n!C_{n-1}^{m-1}$ probable solutions and reduces a large number of abundant solutions.

8	4	2	9	1	6	10	7	3	5

Fig. 1. 'Node Accession' coding of three traveling Salesman and eight cities chromosome.

parameter gene:	8	4	1	7	3	2	5	6
controlling gene:	1	1	2	3	3	1	3	2

Fig. 2. The hierarchical coding of three traveling Salesman and eight cities chromosome.

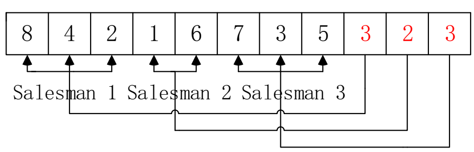

Fig. 3. The suffix coding of three traveling Salesman and eight cities chromosome.

2.2 Self-adaptive genetic operator design

There are many advantages of genetic algorithm while there is also a premature convergence problem, which means converging to partial optimization too early. After population going through several iterations, the individual will have a higher average adaption and it increases exponentially. Due to limited population, these excellent individuals will occupy a lot in population, while other individuals will disappear quickly. This trend is nearly in line with the principal Survival of the Fittest. If it continues developing without manual control then it will result in sole species and an increasing number of near neighbors of parent's population, thus, genetic operator will lost its own value, making algorithm fall into partial optimization. Whether premature convergence problem is obvious, depends on gene cluster, the crossover probability P_c and mutation probability P_m: If P_c is too small, the iterative number will increase and algorithm speed will slow down; whereas P_c is too big, then high-fitness individual will be destroyed quickly; if P_m is too small, it will be hard to produce new individual; whereas P_m is too big, then there will be irregular exhaustion for genetic algorithm. For premature convergence problem of genetic algorithm, it is necessary to design a self-adaptive genetic operator so as to keep population diversity [15].

Self-adaptive genetic operator consists of self-adaptive crossover probability operator and self-adaptive mutation probability operator. The crossover probability P_c has been improved like following formula:

$$P_c = \begin{cases} P_{c1} - \dfrac{(P_{c1} - P_{c2})(f^{'} - f_{avg})}{f_{max} - f_{avg}}, f^{'} \geq f_{avg} \\ P_{c1}, f^{'} < f_{avg} \end{cases} \quad (1)$$

As is shown above, f_{max} is the biggest fitness among the group, f_{avg} is the mean value of group fitness; $f^{'}$ is the bigger fitness between the two crossover individuals; P_{c1} =0.9; P_{c2} =0.6.

The self-adaptive improvement is made to the crossover probability, and the formula is as follows:

$$P_m = \begin{cases} P_{m1} - \dfrac{(P_{m1} - P_{m2})(f_{max} - f)}{f_{max} - f_{avg}}, f \geq f_{avg} \\ P_{m1}, f < f_{avg} \end{cases} \quad (2)$$

f_{max} is the biggest fitness among the group, f_{avg} is the mean value of group fitness; f is the fitness of mutation individuals; P_{m1} =0.1; P_{m2} =0.01.

2.3 Simulated annealing algorithm

The physical background of SA is the physical image and statistical properties during solid annealing process. SA accepts a worse solution than current solution at a certain probability so that the algorithm will jump out of the erroneous zone of local optimum. The procedures of SA can be summarized as follows:

(1) A solution is obtained under a certain temperature T (This is solution can be obtained through optimization of Genetic Algorithm);

(2) If this solution is better than the one under last certain temperature, then this solution will be accepted, otherwise follow (3) ;

(3) Calculate the probability under the temperature T:

$$P = e^{\frac{dE}{KT}} \quad (3)$$

It generates a random number a in interval [0,1]. If $a < P$, then the solution will be accepted, and turn to (1); otherwise give up this solution and turn to (1).

From formula (3) we can see the value of P is getting smaller and smaller with the rise of temperature T. With the proceeding of iterative (the temperature will go down with iterative number), the probability of accepting a bad solution decreases. This is rather similar to crystal's annealing crystallization in nature, hence, SA is realized.

2.4 Modified simulated annealing genetic algorithm

Genetic Algorithm has weaker local search ability. During the later period of Hierarchical Genetic Algorithm, the fitness converges, and less superior individuals are produced. However, Hybrid Simulated Annealing Algorithm can make it jump out of the erroneous zone of local optimum. Due to the compatibility of Genetic Algorithm [16], it is feasible to combine Simulated Annealing Algorithm and Hierarchical Genetic Algorithm to form the modified Simulated Annealing Genetic Algorithm. Now we are going to do the fitness stretch to Hybrid Algorithm:

$$f_i = \frac{e^{\frac{f_i}{T}}}{\sum_{i=1}^{M} e^{\frac{f_i}{T}}} \quad (4)$$

$$T = T_0(0.99^{g-1}) \quad (5)$$

f_i is the fitness of the i the individual; M is the size of population; g is the number of iterations; T is temperature; T_0 is initial temperature. The fitness stretch of Hybrid Simulated Annealing Algorithm has obvious effect. In the initial stage of Genetic Algorithm, individuals with consistent fitness had the similar probability of offspring generation; in the later period, because of significant decrease of temperature [17], the stretch effect got enhanced, so the fitness difference among different individuals is magnified which has made the excellent individuals more superior [18].

Here is the solution procedure of the modified Simulated Annealing Genetic Algorithm.

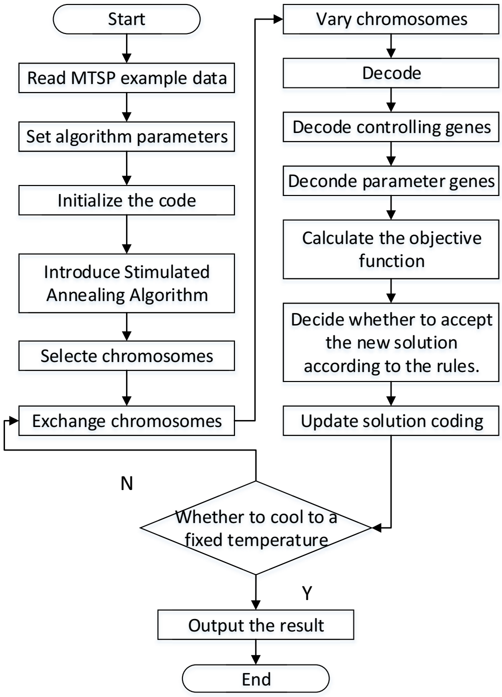

Fig. 4. The flow chart of modified Simulated Annealing Genetic Algorithm.

3 Simulation Results & Analysis

In order to compare modified Simulated Annealing Genetic Algorithm with Simulated Annealing Algorithm, 50 cities are given as simulation objects, and their coordinates in a plane are shown in Table 2. The simulation is divided into 8 groups. The numbers of traveling salesmen and cities in each group are shown in Table 1.

The first group of simulation takes the coordinates of the first 5 groups in Table 2, and uses Hierarchical Genetic Algorithm, Simulated Annealing Algorithm, and modified Simulated Annealing Genetic Algorithm to solve MTSP; each group increases gradually the numbers of traveling salesmen and cities, and the rest can be done in the same manner.

Table 1. Eight simulation variables.

Sequence Number	Number of Salesmans m	Number of Cities n
1	2	5
2	3	10
3	4	15
4	5	20
5	6	25
6	7	30
7	8	40
8	10	50

Table 2. Coordinate values of 50 cities.

City Number	City coordinate		City Number	City coordinate		City Number	City coordinate	
	X	Y		X	Y		X	Y
1	82	84	18	66	54	35	58	22
2	52	30	19	12	23	36	68	18

3	66	41	20	18	36	37	72	74
4	88	15	21	15	60	38	67	68
5	83	40	22	56	45	39	78	53
6	62	64	23	20	20	40	18	49
7	83	63	24	30	71	41	22	82
8	76	88	25	45	85	42	36	17
9	30	82	26	51	72	43	43	30
10	87	51	27	32	41	44	62	33
11	27	19	28	75	46	45	68	30
12	83	89	29	62	82	46	75	28
13	12	89	30	39	68	47	75	58
14	47	33	31	12	43	48	88	41
15	67	47	32	12	78	49	83	72
16	23	56	33	35	31	50	11	55
17	18	72	34	46	18			

3.1 Simulation results of 10 traveling salesmen and 15 cities

Three algorithms are used to simulate MTSP (m=10, n=50) and these are all the convergences and solutions.

The algorithm convergence of Hierarchical Genetic Algorithm is in Figure5, and the solution is in Figure6.

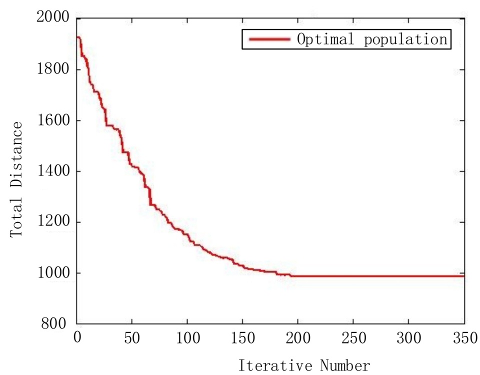

Fig. 5. The iteration curve of Hierarchical Genetic Algorithm optimum MTSP(m=10, n=50).

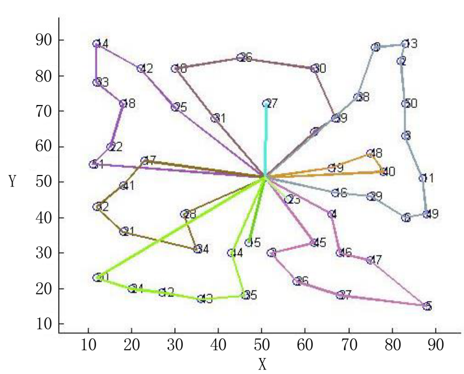

Fig. 6. The solution of Hierarchical Genetic Algorithm optimum MTSP(m=10，n=50).

The algorithm convergence of Simulated Annealing Genetic Algorithm is in Figure7, and the solution is in Figure8.

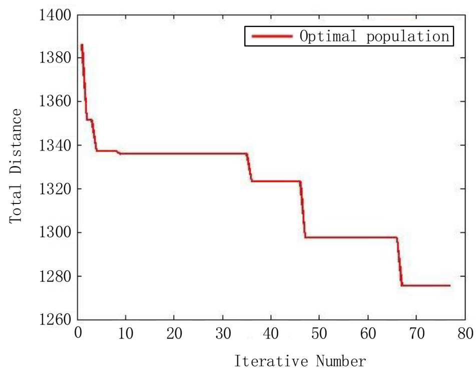

Fig. 7. the iteration curve of Simulated Annealing Genetic Algorithm optimum MTSP(m=10, n=50).

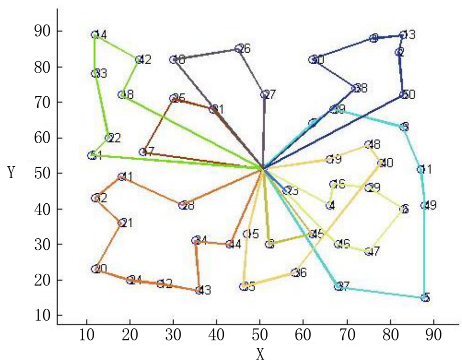

Fig. 8. The solution of Simulated Annealing Genetic Algorithm optimum MTSP (m=10, n=50).

The algorithm convergence of modified Simulated Annealing Genetic Algorithm is in Figure9, and the solution is in Figure 10.

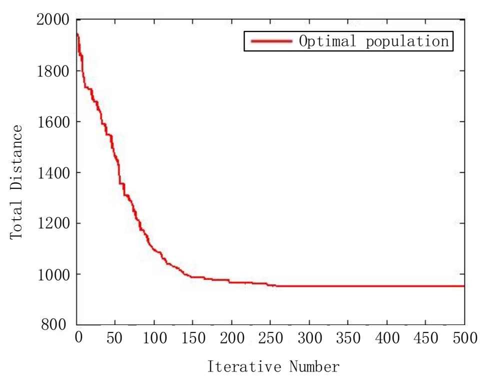

Fig. 9. the iteration curve of modified Simulated Annealing Genetic Algorithm optimum MTSP (m=10, n=50).

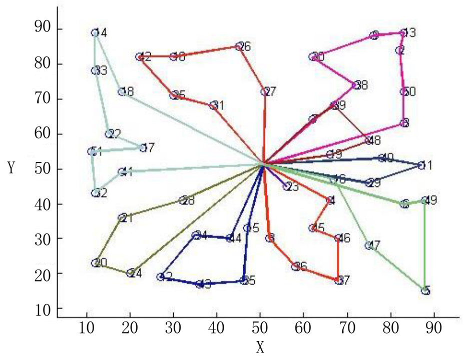

Fig. 10. The solution of modified Simulated Annealing Genetic Algorithm optimum MTSP (m=10，n=50).

3.2 Analysis of simulation results

By the comparison of the simulation results of the three algorithms in Table 3 and Figure11, it is clear that Hierarchical Genetic Algorithm is able to converge to a better solution, while Simulated Annealing Algorithm iterates faster. However, modified Simulated Annealing Genetic Algorithm can sooner achieve a better global optimum solution. The reasons are: the suffix structure design of chromosome reduced the space of the solution, the self-adaptive genetic operator and double crossover and mutation improved ‘premature convergence problem’; the introduction of Simulated Annealing Algorithm stretched the fitness and enhanced the local search ability [19].

Table 3. The comparison of the simulation results of the three algorithms.

	m=2 n=5	m=3 n=10	m=4 n=15	m=5 n=20	m=6 n=25	m=7 n=30	m=8 n=40	m=10 n=50
Hierarchical Genetic Algorithm（HGA）	198	316	456	512	612	692	828	998
Simulated Annealing Algorithm（SA）	185	323	556	735	776	821	996	1270
Modified Simulated Annealing Genetic Algorithm (MSAGA)	193	299	413	474	576	668	778	956

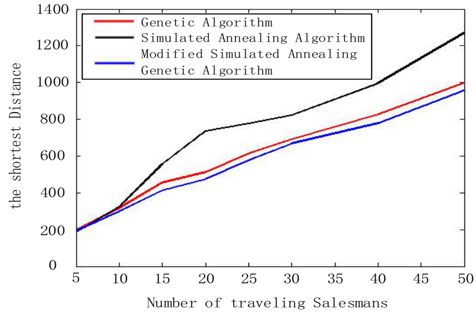

Fig. 11. The comparison of the simulation results of the three algorithms.

4 Conclusion

At present, there are numerous researches on MTSP. The situation of MTSP is complex, the solving of MTSP is difficult, and the research achievements of MTSP are inadequate [20].This thesis adopts a modified Simulated Annealing Genetic Algorithm to solve MTSP. The algorithm has following features:

(1) A suffix structure design of chromosome was put forward to reduce the space of the solution and eliminate most redundant solution;

(2) A self-adaptive genetic operator is devised for double crossover and mutation which can both remain population diversity and algorithm convergence;

(3) Simulated Annealing Algorithm and Hierarchical Genetic Algorithm are combined, and stretch the fitness, enhance the local search ability so as to achieve a better global optimum solution.

Acknowledgements

Finally, I would like to extend my sincere gratitude to my supervisor, Sheng Li, for his instructive advice and useful suggestions on my work. I am deeply grateful of his help in the completion of this work.

And, this work was supported by the National Natural Science Foundation of China (61673219), the Jiangsu Six Talents Peaks Project of Province (XNYQC-CXTD-001) and the Tianjin Major Projects of Science and Technology (15ZXZNGX00250).

References

1. Shim, V. A.,Tan, K. C.,Cheong, C. Y. et al.A Hybrid Estimation of Distribution Algorithm with Decomposition for Solving the Multiobjective Multiple Traveling Salesman Problem.IEEE transactions on systems, man and cybernetics, Part C. Applications and reviews: A publication of the IEEE Systems, Man, and Cybernetics Society,2012,42(5):682-691.
2. Daibo Liu,Mengshu Hou,Hong Qu et al.A simple model for the multiple traveling salesmen problem with single depot and multiple sink.COMPEL: The international journal for computation and mathematics in electrical and electronic engineering,2013,32(2):556-574.
3. Marco Casazza,Alberto Ceselli,Marc Nunkesser et al.Efficient algorithms for the double traveling salesman problem with multiple stacks.Computers & operations research,2012,39(5):1044-1053.
4. Cheang, B.,Gao, X.,Lim, A. et al.Multiple pickup and delivery traveling salesman problem with last-in-first-out loading and distance constraints.European Journal of Operational Research,2012,223(1):60-75.
5. Marco Casazza,Alberto Ceselli,Marc Nunkesser et al.Efficient algorithms for the double traveling salesman problem with multiple stacks.Computers & operations research,2012,39(5):1044-1053.
6. Carrabs, F.,Cerulli, R.,Speranza, M.G. et al.A branch-and-bound algorithm for the double travelling salesman problem with two stacks.Networks: An International Journal,2013,61(1):58-75.
7. Estevez-Fernandez A,Borm P,Hamers H et al.On the core of multiple longest traveling salesman games.European Journal of Operational Research,2006,174(3):1816-1827.
8. Elad Kivelevitch,Kelly Cohen,Manish Kumar et al.A Market-based Solution to the Multiple Traveling Salesmen Problem.Journal of Intelligent & Robotic Systems: Theory & Application,2013,72(1):21-40.
9. Angel Felipe,M. Teresa Ortuno,Gregorio Tirado et al.The double traveling salesman problem with multiple stacks: A variable neighborhood search approach.Computers & operations research,2009,36(11):2983-2993.
10. Qu H, Yi Z, Tang H J. A columnar competitive model for solving multi-traveling salesman problem . Chaos, Solitons and Fractals. Chaos, solitions & Fractals, 2007,31(4):1009-1019.
11. Laporte G, Nobert Y. A cutting planes algorithm for the m-salesmen problem . Journal of the Operational Research Society, 1980, 31:1017-1023.
12. Russell R A. An effective heuristic for the m-tour traveling salesman problem with some side conditions . Operations Research, 1977, 25(3):517-524.
13. Yonggang Peng,Xiaoping Luo,Wei Wei et al.A New Fuzzy Adaptive Simulated Annealing Genetic Algorithm and Its Convergence Analysis and Convergence Rate Estimation.International Journal of Control, Automation, and Systems,2014,12(3):670-679.
14. Hoseini, P.,Shayesteh, M.G..Efficient contrast enhancement of images using hybrid ant colony optimisation, genetic algorithm, and simulated annealing.Digital Signal Processing,2013,23(3):879-893.
15. Min Dai,Dunbing Tang,Adriana Giret et al.Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm.Robotics and Computer Integrated Manufacturing: An International Journal of Manufacturing and Product and Process Development,2013,29(5):418-429.
16. Ying Xu,Rong Qu,Renfa Li et al.A simulated annealing based genetic local search algorithm for multi-objective multicast routing problems.Annals of operations research,2013,206(Jul.):527-555.
17. Symone Soares,Carlos Henggeler Antunes,Rui Araujo et al.Comparison of a genetic algorithm and simulated annealing for automatic neural network ensemble development.Neurocomputing,2013,121(Dec.9):498-511.
18. Xiaorong Xie*, Qirong Jiang, Yingduo.Damping multimodal subsynchronous resonance using a static var compensator controller optimized by genetic algorithm and simulated annealing.European transactions on electrical power,2012,22(8):1191-1204.
19. A.F. Crossland,D. Jones,N.S. Wade et al.Planning the location and rating of distributed energy storage in LV networks using a genetic algorithm with simulated annealing.International journal of electrical power and energy systems,2014,59(Jul.):103-110.
20. Meei-Yuh Ku,Michael H. Hu,Ming-Jaan Wang et al.Simulated annealing based parallel genetic algorithm for facility layout problem.International journal of production research,2011,49(6/8):1801-1812.