Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/070637

Particle Swarm Optimization Algorithm for Multi-salesman Problem with Time and Capacity Constraints

Shanchen Pang[1,2,*], *Tan Li*[1], *Feng Dai*[1] *and Meng Yu*[2]

[1] College of Information Science & Engineering, Shandong University of Science & Technology, 266590 Qingdao, China
[2] Computer Science Department, School of Engineering, Virginia Commonwealth University, 23284 Richmond, USA

Received: 1 Dec. 2012, Revised: 2 Mar. 2013, Accepted: 10 Mar. 2013
Published online: 1 Nov. 2013

Abstract: Classic multiple traveling salesman problem (MTSP) requires to find the k closed circular paths which minimize the sum of the path lengths, and each vertex is visited only once by a salesman. This paper presents an optimized model for the balanced Multiple-salesman Problem with time and capacity constraints, it requires that a salesman visits each vertex at least once and returns to the starting vertex within given time. The balanced MSP is more widely used than MTSP. We describe a particle swarm optimization algorithm for balanced MSP and explain its simple application in the Post Office Scheduling Problem. We demonstrate the effectiveness of the model and algorithm through a case study.

Keywords: MTSP, Multiple-salesman Problem, Time and Capacity Constraints, Particle Swarm Optimization Algorithm.

1. Introduction

In the classic multiple traveling salesman problem (MTSP), given n cities and k salesmen, we need to find k closed circular paths that minimize the sum of the path lengths. An MTSP can be viewed as a multiple traveling salesman problem (TSP) with closed loops, i.e., an MTSP can be converted to a single TSP problem. The TSP is an NP-complete combinatorial optimization problem. Both TSP and MTSP are widely used in network routing, vehicle routing problems, etc. Many researchers developed optimization algorithms based on dynamic linear programming theory. These algorithms can be divided into two categories [1], One category has local heuristic search algorithms, such as k-opt algorithm, LK algorithm, and LKH algorithm. The other category has artificial intelligence algorithms, such as ant colony algorithms [2,3], genetic algorithms [4-7], neural network algorithms [8].

This paper presents an optimized model for the balanced Multiple-salesman Problem (MSP) with time and capacity constraints that can be described as: given n cities and k salesmen, find the k closed circular paths that minimize the sum of the path lengths. It requires that each one city is visited at least once by a salesman and return the starting city within given time. Because of the time and capacity constraints, it needs to ensure that each path is as equal as possible. The balanced MSP is a problem often occurs in real cases. However, there is no effective way to solve it currently. In this paper, we describe a particle swarm optimization (PSO) algorithm for the balanced MSP and demonstrate its application in the Post Office scheduling problem. Our example shows the effectiveness of the model and algorithm.

2. Multiple-salesman Problem

Definition 1 (the optimal salesman circular path): given a weighted graph $G=(V,E,w)$, where V is the set of vertexes, E is the set of edges, w is the weight function, the closed circular path with all vertices visited at least once, and the minimum sum of weight of path, is called the optimal salesman circular path.

In fact, a solution to the salesman problem is the optimal Hamilton (H) circle, but in some special cases, for example, under different time and capacity constraints, a solution to the best salesman problem may be better than the optimal H circle.

Example 1. In Figure 1, the optimal H circle is ABCA with length 5. For the best salesman problem, the solution ACBCA has length 4.

* Corresponding author e-mail: shanchenpang2008@gmail.com

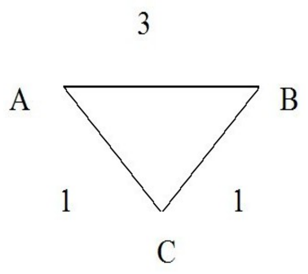


Figure 1. Example 1 Sketch.

Definition 2 (Multiple-salesman Problem MSP): In a weighted graph G, given a vertex, to find K closed circular paths that leave from the given vertex and ultimately be back to the starting vertex

(1) K closed circular paths cover all the vertices of the graph G.

(2) The sum of weights of K closed circular paths is minimal.

This problem is called Multiple-salesman Problem [1].

2.1. Model of MSP

Based on the definition of MSP given above, the model of MSP can be described as: given a weighted graph $G = (V,E,w)$, V is the vertex set, $V = \{0,1,\cdots,n\}$, 0 is the starting vertex, E is the set of edges, $w = \{d_{ij}|(i,j)\in E\}$ is the set of weight of edges (it expresses the weight between vertexes, such as distance, time, cost, etc.). The vertex set V of G can be divided into K subsets $V_1,V_2,\cdots,V_k$, each subset contains the starting vertex. T_i is the sum of weight that is the optimal salesman circular path visited set V_i and satisfies the following conditions:

$$min \sum_{i=1}^{k} T_i$$

$$\bigcup_{i=1}^{k} V_i = V$$

2.2. Balanced MSP

Equilibrium problem is a multi-objective problem. In the practical application, to balance the tasks is very important. The balanced task of MSP is divided into two cases: (1) the difference between numbers of vertices that salesmen visit should be as little as possible, (2) the sum of weight of the path that a salesmen visit should have no big difference from others. The time and capacity constraints in this paper cover both the two cases.

2.3. Optimized Model of the Balanced MSP

For the contradictions between the two balance cases proposed in 2.2. In normal, the shortest path will not be balanced. In order to consider the both cases, we will establish the optimized model of the balanced MSP.

Given a weighted graph G, we will add K virtual nodes $n+1,...,n+k$ into V to obtain the K salesman circles. V changes into $\{0,1,...,n,n+1,...,n+k\}$, and the weight is defined as:

$$d_{n+j,0} = d_{0,n+j} = d_{n+i,n+j} = d_{n+j,n+i} = 0, \quad (j = 1,2,\cdots,k).$$
$$d_{n+i,j} = d_{j,n+i} = d_{0,j} = 0, \quad (i = 1,2,\cdots,k, j = 1,2,\cdots,n).$$

We introduce a decision variable, defined as:

$$x_{ij} = \begin{cases} 1, & (i,j) \ is \ visited, \\ 0, & else. \end{cases}$$

The optimized model of the balanced MSP can be expressed as:

$$min \max_{p\in\{1,2,\cdots,k\}} \Sigma_{i,j\in V_P} d_{i,j} * x_{i,j} \tag{1}$$

$$s.t. \ f_1 = \prod_{i=1}^{n+k}(\sum_{j=1}^{n+k} x_{i,j}) \geq 1. \tag{2}$$

Equation (1) ensures to obtain the shortest path form the maximum path set in order to balance two cases proposed in 2.2, (2) ensures that a salesman visits each vertex at least once.

3. PSO Algorithm for Optimized Model of the Balanced MSP

3.1. Particle swarm optimization algorithm

In 1995, Eberhan and Kennedy proposed the particle swarm optimization algorithm[9,10]. The basic idea of PSO algorithm is to find the optimal solution through collaboration and information sharing between individuals in the group. PSO is initialized to a group of random particles, and let the scale of the group be S. Each particle represents a candidate solution of the solution space, $X_i = (x_{i1},x_{i2},\cdots,x_{in})$ represents the location of the particle i in the n-dimensional space, $V_i = (v_{i1},v_{i2},\cdots,v_{in})$ is the flight speed.

According to the optimization goal, to determine the fitness function, in each time iteration, the particle use dynamical tracking the two extreme values to update of its own, one is the particle's own flight experience that iterative from the initial to the current best position *pbest*, the other is the best position that all particles *gbest* found so far. For the kth time iteration, the speed and position of the particle are expressed by the following formulas[11]:

$$v_{ij}^{k+1} = w * v_{ij}^{k} + c_1 rand(1) * (pbest_{ij} - x_{ij}^{k}) + c_2 rand(2) * (gbest_{ij} - x_{ij}^{k}) \quad (3)$$

$$x_{ij}^{k+1} = v_{ij}^{k+1} + x_{ij}^{k}, (i = 1, 2, \cdots, S, j = 1, 2, \cdots, n) \quad (4)$$

Where, $rand(1)$ and $rand(2)$ are random distribution in $[0,1]$, and c_1 and c_2 are the learning factors, and w is the dynamic inertia weight, decreasing linearly with the iterative, respectively. We usually select the following formula[12]:

$$w = w_{max} - \frac{w_{max} - w_{min}}{T_{max}} * T$$

T is the iteration number at the current time step; T_{max} is the maximum number of iterations. In each iteration, a number of particles, which have better fitness value, are chosen as the leaders. The leaders are used to seek the construct of creativity and innovation, that is, particles find the optimal solution switch the two statuses between individual extreme value and global extreme value in the solution space, and until the iteration conditions are satisfied.

3.2. PSO algorithm for optimized model of the balanced MSP

According to optimized model of the balanced MSP mentioned in 2.3, each vertex is visited at least once by the salesman, and the times of visitation should be as little as possible, we assume that the dimension of each particle is 2n, the number of initial particles is 20, and each particle cover the vertex set V.

Considering the status of a particle is affected or determined by the value of the objective function. We set the objective function as fitness function. That is,

$$f_2 = \max_{p \in \{1,2,\cdots,k\}} \Sigma_{i,j \in V_P} d_{i,j} * x_{i,j}.$$

Let learning factors $c_1 = c_2 = 2$, and inertia weight w is linearly decreasing from 0.9 to 0.4 according to Equation (3) and Equation (4), and the times of iterations is 500. The specific steps are shown as follows:

1. Initialize the speed and position of all particles in the population.
2. Calculate the fitness function f_2 of each particle.
3. If the fitness of the particle is less than its individual extreme value *pbest*, then update *pbest*.
4. Elect a particle whose fitness function is the smallest as *pbest* in current population.
5. If the iterative conditions are satisfied, then output *gbest* and exit. Otherwise, update the speed and position of particle according to the formula (3) and (4), and the updated location divided by $n + k$, and round the remainder. Finally, calculate f_1, if $f_1 = 0$ and $x_i^{k+1} = x_i^k$, return to step 2.

Algorithm flowchart is shown in Figure 2.

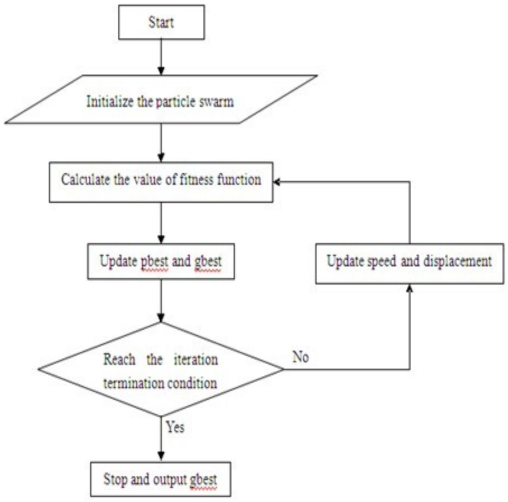


Figure 2. Algorithm flowchart.

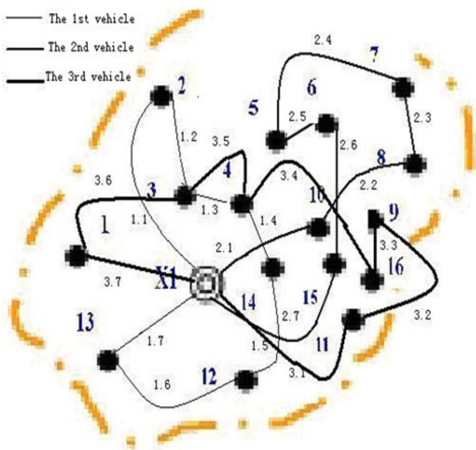


Figure 3. Postal routes planning and post vehicles scheduling program.

4. Case Study

Example 2. A post office X_1 has sixteen branches $Z_1, Z_2, \cdots, Z_{16}$, shown in Figure 3. Suppose the post vehicles depart from X_1 at 09:00 to the sixteen branches $Z_1, Z_2, \cdots, Z_{16}$ to send and receive parcels, and return at 15:00. If the maximum load of each post vehicles is 65 bags of parcels at most, now the question is how to get an optimal transport strategy in order to improve the postal transportation efficiency? Table 1 shows the amount of the sending and accepting parcels. In addition, vacancy rate=(the maximum loading of parcels-the actual loading of parcels of the post vehicle)/the maximum load of parcels, the reduced income due to the vacancy rate is (vacancy rate×2/km).

4.1. Problem analysis

Considering the cost of post vehicles transportation, the vacancy rate is a key factor in the efficiency of postal transportation. Therefore, the goal of the solving is to conserve the globe vacancy rate to the minimum, in condition of meeting the time and capacity constraints of the post vehicles. In addition, it also ensures that each branch is visited by one post vehicle at least once to unload and load its parcels. The number of parcels after being unloaded and loaded must not exceed the maximum capacity of the post vehicle. We can use PSO Algorithm for Optimized Model of the Balanced MSP to obtain the plan of postal routes and the post vehicle scheduling program.

Table 1. the amount of parcels in each branch.

branch office	Amount of parcels sent from X_1 to Z_i	Amount of parcels sent from Z_i to X_1
Z_1	10	9
Z_2	15	14
Z_3	6	5
Z_4	9	10
Z_5	13	9
Z_6	6	10
Z_7	11	13
Z_8	4	9
Z_9	13	15
Z_{10}	17	9
Z_{11}	11	6
Z_{12}	2	7
Z_{13}	11	13
Z_{14}	21	15
Z_{15}	13	10
Z_{16}	14	16

4.2. The balanced MSP model

According to the analysis in 4.1, we set the loss income due to the vacancy rate as the optimized objective function, and set time and capacity as constraints functions, the model were defined as follows:

$$min\ \ Z(x)=\sum_{k\in S}\sum_{i\in N}\sum_{j\in N}2\cdot\frac{65-V_{ijk}}{65}d_{ij}\cdot X_{ijk}$$

$$s.t.\quad V_{ijk}\leq 65,\quad \forall k\in S,\forall i\in N,\forall j\in N \tag{5}$$

$$V_{i_m jk}=V_{i_{m-1}jk}+w_{i_{m-1}}-u_{i_{m-1}} \tag{6}$$

$$\frac{\sum_{i\in N}\sum_{j\in N}d_{ij}\cdot X_{ijk}}{30}+\frac{1}{12}\sum_{i\in N}y_{kj}\leq 6 \tag{7}$$

Where, $N=\{0,1,2,\cdots,16\}$ represents $X_1,Z_1,Z_2,\cdots,Z_{16}$ respectively; $S=\{0,1,2,\cdots\}$ indicates the number of post vehicles; V_{ijk} represents the number of parcels(bags) of the kth post vehicle after unloading and loading parcels in the branch Z_i; d_{ij} represents the distance between the branch Z_i and Z_j (km);

$$X_{ijk}=\begin{cases}1, & if\ the\ k^{th}\ postal\ vehicle\ departs\ from \\ & the\ branch\ Z_i\ to\ branch\ Z_j\\ 0, & else\end{cases}$$

w_i represents the number of parcels that branch Z_i received(bags); u_i represents the number of parcels unloaded to the branch Z_i(bags);

$$y_{ki}=\begin{cases}1, & if\ the\ k^{th}\ postal\ vehicle\ unload\\ & and\ load\ parcels\ in\ Z_i\\ 0, & else\end{cases}$$

The objective function represents the loss income due to the vacancy rate. Considering of the constraints, formula (5) guarantees that the number of parcels of post vehicle must not exceed its maximum capacity; formula (6) presents the change in the number of parcels when the post vehicle visited a branch; formula (7) guarantees that all post vehicles return to the post office X1 within the time window.

4.3. Result based on particle swarm optimization

For the planning model above in 4.2, Using PSO algorithm, the answer is it needs three post vehicles at least meeting the demand for the transport. In order to improve the postal transport efficiency, the details of the postal routes planning and the post vehicles scheduling program are as follows:
The route which the first post vehicle visited:

$$X_1\to Z_2\to Z_3\to Z_4\to Z_{14}\to Z_{12}\to Z_{13}\to X_1$$

The route which the second post vehicle visited:

$$X_1\to Z_{10}\to Z_8\to Z_7\to Z_5\to Z_6\to Z_{15}\to X_1$$

The route which the third post vehicle visited:

$$X_1\to Z_{11}\to Z_9\to Z_{16}\to Z_4\to Z_3\to Z_1\to X_1$$

In case of this scheduling, the minimum loss income due to the vacancy rate is 48.2.

The post vehicles scheduling program is shown in figure 3.

5. Related Work

PSO algorithm is one of the most attention nature-inspired algorithms recently. This algorithm is used to solve many complex issues, and to get better results. Rajesh Kumar et al. presented a new multi-agent based hybrid particle swarm optimization technique applied to the economic power dispatch, the algorithm integrates the deterministic search, the Multi-agent system, PSO algorithm and the bee decision-making process [13]. Yannis Marinakis and Magdalene Marinaki analyzed the probabilistic traveling salesman problem and proposed a new hybrid algorithmic nature inspired approach based on PSO to solve the problem[14]. Amir Robati et al. extended PSO algorithm which is essentially based on balanced fuzzy sets theory [15]. The extension algorithm has the similar ideas with our algorithm. Hanhong Zhu et al. presented a meta-heuristic approach to portfolio optimization problem using PSO technique[16]. Yen-Far Liao et al. adopt a two-step method to improve PSO to solve TSP, the first phase includes Fuzzy C-Means clustering, and the second phase proposes Genetic-based PSO procedure to TSP with better efficiency[17].

6. Conclusions and future work

This paper mainly discusses the postal routes planning and the post vehicles scheduling program. Under specific time and capacity constraints, we present an optimized model for the balanced Multiple-salesman Problem, and described a particle swarm optimization algorithm for the balanced MSP and demonstrated its simple application in the Post Office scheduling problem. A case study shows the effectiveness of the model and algorithm. We will analyze the complexity of the algorithm and compare it with other algorithms in our future work.

Acknowledgement

This work is supported by the National Grand Fundamental Research 973 Program of China (No.2010CB328105), the National Natural Science Foundation of China (No.91018007, 60970001, 60874036, 61170183), the Scientific Research Foundation for the Excellent Middle-Aged and Youth Scientists of Shandong Province of China (No.BS2011DX027, BS2011SW025), the Outstanding Youth Foundation of Shandong University of Science and Technology of China(No.2011KYJQ104), the Technology development projects of Qingdao City of China(No. 12-1-4-6-(7)-jch).

References

[1] Baraglia R J I, and Hidalgo R Perego, A hybrid heuristic for the traveling salesman problem. IEEE Transaction on Evolutionary Computation, **5**, 613-622 (2001).

[2] Dorigo M, Maniezzo V, and Colorni A., The ant system: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man and Cybernetics, Part B., **26**, 29-42 (1996).

[3] Duan H B,Wang D B, and Yu X F., Research on the optimum configuration strategy for the adjustable parameters in ant colony algorithm. Journal of Communication and Computer, **2**, 32-35 (2005).

[4] Guo T, and Michalew icz Z., Inver-Over operator for the TSP. Proceeding of the 5th Parallel Problem Solving form Nature, 803-812 (1998).

[5] Pataki, G., Teaching integer programming formulations using the traveling salesman problem. SIAM Review, **45**, 116-123(2003) .

[6] Merz P, and Freisleben B. Genetic local search for the TSP:New results. Proceeding of the 1997 IEEE International Conference on Evolutionary Computation, 259-264 (1997).

[7] Merz P, and Freisleben B., Memetic algorithms for the traveling salesman problem. Complex System, **13**, 297-345 (2001).

[8] Lawler, E.; Lenstra, J.; Rinnooy Kan, A.; and Shmoys, D. The traveling salesman problem: A guided tour of combinatorial optimization. New York: Wiley, (1985).

[9] Kennedy, J., and Eberhart, R., Particle swarm optimization. Proceeding of the IEEE International Conference on Neural Networks, 1942-1948 (1995).

[10] Eberhart R, and Kennedy J., A new optimizer using particle swarm theory. Proceedings of the 6th International Symposium on Micro Machine and Human Science, 39-43 (1995).

[11] Clerc M. The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization. In: Proc. of the ICEC. Washington, 1951-1957 (1999).

[12] Bin, Wei; Qinke, Peng; Jing, Zhao; Xiao, Chen, A binary particle swarm optimization algorithm inspired by multi-level organizational learning behavior. European Journal of Operational Research, **219**, 224-233 (2012).

[13] Rajesh Kumar, Devendra Sharma, Abhinav Sadu, A hybrid multi-agent based particle swarm optimization algorithm for economic power dispatch. Electrical Power and Energy Systems, **33**, 115-123 (2011).

[14] Yannis Marinakis,Magdalene Marinaki, A Hybrid Multi-Swarm Particle Swarm Optimization algorithm for the Probabilistic Traveling Salesman Problem. Computers and Operations Research, **37**, 432-442 (2010).

[15] Amir Robati, Gholam Abbas Barani, Hossein Nezam Abadi Pour et al., Balanced fuzzy particle swarm optimization. Applied Mathematical Modelling, **36**, 2169-2177 (2012).

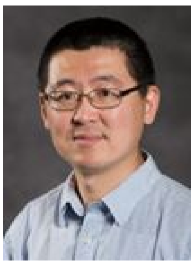
[16] Hanhong Zhu, Yi Wang et al., Particle Swarm Optimization (PSO) for the constrained portfolio optimization problem. Expert Systems With Applications, **38**, 10161-10169 (2011).

[17] Yen-Far Liao, Dun-Han Yau, Chieh-Li Chen, Evolutionary algorithm to traveling salesman problems. Computers and Mathematics with Applications, **64**, 788-797 (2012).

Shanchen Pang received the Ph.D. in computer science from Tongji University, Shanghai, China, in 2008. He is currently a associate professor in computer Engineering and Science at Shandong University of Science & Technology, and Deputy director of Chinese Professional Committee of Computer Society of Petri nets. His research interests include the Petri net theory and application, algorithm design, distributed concurrent system analysis and verification.

Tan Li now study the Master of computer application technology in Shandong University of Science and Technology from 2011. She is currently doing some research about system simulation and workflow service as an internship in Peking University. Her research interests include Petri net and BPM.

Feng Dai received the bachelor degree in College of Information Science and Engineering of Shandong University of Science and Technology in Jun. 2012. She has also got admission to graduate school of Alma Master in 2011 and will study for the master's degree from 2012. Her research interests include Petri net and workflow.



Meng Yu received the Ph.D. in computer science ftom Nanjing University, Nanjing, China, in 2001. He is currently a associate professor in Department of Computer Science at Virginia Commonwealth University. His research interests include computer and network security, cloud computing, self-healing and survivable systems.