Advance Journal of Food Science and Technology 10(8): 563-567, 2016
DOI: 10.19026/ajfst.10.2184
ISSN: 2042-4868; e-ISSN: 2042-4876

Submitted: May 16, 2015 Accepted: July 2, 2015 Published: March 15, 2016

Research Article

The Particle Swarm Algorithm for Farmland Food Harvester Routing Optimization with Multiple-depot and Time Windows

[1]Qiang Song, [1]Weiwei Zhang and [2]Lingxia Liu
[1]Anyang Institute of Technology,
[2]School of Software Engineering, Anyang Normal University, Anyang, 455000, China

Abstract: Multi-depot farmland food harvester routing problem with time windows is a variation of the farmland food harvester routing problem constrained by multi-depot and time windows, which is a typical NP-hard problem. Particle swarm optimization with a particle coding method is designed to solve the problem. The mathematic mode is established and the solution algorithm is developed. The simulation results of example indicate that the algorithm has faster search speed and stronger optimization ability than the genetic algorithm and ant colony algorithm.

Keywords: Farmland food harvester, multi-depot, particle swarm optimization, route problem, time windows

INTRODUCTION

Farmland food harvester Routing Problem (VRP) was firstly proposed by Dantzig and Ramser (1959). In reality many logistics farmland food harvester optimization problem can be come down to farmland food harvester routing problem with windows and multiple-depot. The least path cost is needed to deliver a specified number of goods within the required time to customer points and finally the farmland food harvesters back to the depot. Although the application of this problem is very widely, "multiple-depot" and "time windows" these two constraints respectively applying in the aspects of space and time, makes the solving become more complex which originally had the difficulty of a NP-hard problem, only when the problem size is smaller, it is likely to get its exact solutions; it is difficult to obtain the optimal solution especially for big dimensional problems. In recent years, the heuristic optimization algorithm, such as genetic algorithm and ant colony algorithm, obtains the preliminary application in solving this kind of problems (Chaos *et al*., 1983; Yuanfeng, 2008; Meijun *et al*., 2008), but the solution's accuracy and efficiency need improvement.

Particle Swarm Optimization (PSO) is a newly appearing bionic algorithm which imitates the bird's flying for food, with the advantages of less number of individuals, simple calculation, fast convergence speed and easy to implement and in the farmland food harvester routing optimization problem this algorithm is implemented and obtains good effect (Kennedy and Eberhart, 1995; Zhixiong, 2009; Yuanbiao and Guangqing, 2007).

In this study, we utilizes the particle swarm algorithm to optimize the Multiple-Depot farmland food harvester Route Problem with Time Windows (MDVRPTW). Through the experiments proof, this algorithm has obtained the good effect to reduce computing time and avoid algorithm precocity.

METHODOLOGY

The description and mathematical model of MDVRPTW: The multiple-depot farmland food harvester routing problem with time windows can be described as: there are M depots, which respectively own $K_m = (1, 2, \ldots, M)$ farmland food harvesters with capacity q, now it is known the transportation distance from customer i to customer j is d_{ij} (here the distance is used to replace the transportation cost). N customers need to do goods distribution, the speed of the farmland food harvester is v, the goods must be served within time of l_i $(1 = 1, 2, \ldots, N)$, the goods demand of customer i is $g_i = (1, 2, \ldots, N)$ and $g_i<q$, each customer can be served by farmland food harvesters in any one depot, but only one truck can serve once, after every truck completes the distribution task it must return back to the original depot. This process requests to arrange the appropriate farmland food harvester driving route, make the farmland food harvesters of each depot can meet all the needs of customers and make the total traveling distance minimum.

Here the customer codes are 1, 2, …, N, the depots codes are N+1, N+2, …, N+M, t_{ik}, is the time that farmland food harvester k to customer i; t_{ijk} is the driving time from node i to node j of farmland food harvester k; the variables are defined as expression (1).

Corresponding Author: Qiang Song, Anyang Institute of Technology, Anyang, 455000, China

Apparently the set of all the maximum terms consist of m atoms is surely contains 2 m maximum terms. Therefore, it is only need to compute the number of distinct maximum terms can be deduced from the clause set that we can determine its satisfiability. In addition, when counting the number of the maximum terms that can be deduced from the clause set, we can use the inclusion-exclusion principle presented below:

$$x_{ij}^{mk} = \begin{cases} 1, & \textit{the vehicle k in depot m driving} \\ & \textit{from customer i to customer j} \\ 0, & \textit{otherwise} \end{cases} \quad (1)$$

Thus, the farmland food harvester route mathematical model of multiple-depot is shown in expression (2):

$$f = \min \sum_{i=1}^{N+M} \sum_{j=1}^{N+M} \sum_{m=1}^{M} \sum_{k=1}^{K_m} d_{ij} x_{ij}^{mk} \quad (2)$$

The constraints conditions are set as expression (3) to (8):

$$\sum_{i=1}^{N} \sum_{k=1}^{K_m} x_{ij}^{mk} \le K_m$$
$$(i = m \in \{N+1, N+2, ..., N+M\}) \quad (3)$$

$$\sum_{i=1}^{N+M} \sum_{m=1}^{M} \sum_{k=1}^{K_m} x_{ij}^{mk} = 1 \ (i \in \{1,2,...,N\}) \quad (4)$$

$$\sum_{j=1}^{N+M} \sum_{m=1}^{M} \sum_{k=1}^{K_m} x_{ij}^{mk} = 1 \quad (j \in \{1,2,...,N\}) \quad (5)$$

$$\sum_{i=1}^{N} x_{ij}^{mk} = \sum_{j=1}^{K} x_{ij}^{mk} \le 1 \quad (6)$$

$$(i = m \in \{N+1, N+2, ..., N+M\},$$
$$k \in \{1,2,...,K_m\})$$

$$t_{ijk} = \frac{d_{ij}}{v} (i, j \in \{1,2,...,N\}) \quad (7)$$

$$t_{ik} = t_{ik} + t_{ijk} x_{ij}^{mk} \quad (8)$$

$$(i, j \in \{1,2,...,N\}, k \in \{1,2,...,K_m\})$$
$$t_{ik} \le l_i, i, j \in \{1,2,...,N\}, k \in \{1,2,...,K_m\} \quad (9)$$

$$\sum_{i=1}^{N} g_i \sum_{j=1}^{N+M} x_{ij}^{mk} \le q$$
$$(m \in \{N+1, N+2, ..., N+M\}, \ \ k \in \{1,2,...,K_m\}) \quad (10)$$

$$\sum_{i=N+1}^{N+M} x_{ij}^{mk} = \sum_{j=N+1}^{N+M} x_{ij}^{mk} = 0$$
$$(i = m \in \{N+1, N+2, ...N+M\}, k \in \{1,2,...K_m\}) \quad (11)$$

In the model, expression (2) is the objective function, which is the lowest transportation cost for farmland food harvester; expression (3) limits the sum of farmland food harvesters from each depot cannot exceed the sum of farmland food harvesters owned by this depot; expression (4) and (5) ensure that each customer can only be served once by a truck; expression (6) ensures that each farmland food harvester starts from their own depot and returns back to this depot; expressions (7)-(9) represent the time constraints of the farmland food harvester driving; expression (10) represents the farmland food harvester load should not exceed its load capacity; expression (11) ensures that the farmland food harvester can't drive from one depot to another depot.

The particle swarm algorithm:

Introduction particle swarm algorithm: Particle Swarm Optimization algorithm (PSO) is a new algorithm based on swarm intelligence theory proposed by psychologist Kennedy and electrical engineer Eberhart in 1995 (Kennedy and Eberhart, 1995), who were inspired by flying bird's behavior of looking for food in the real world[5]. This algorithm adopts the simple velocity-displacement model, under the guide of the fitness value function information to realize the optimal search of each particle, each particle follows the current optimal particle to do global search in the solution space. The optimal particle includes two types: individual optimal particle and group optimal particle. Each particle itself in the process of iteration finds the optimal solution, which is the individual optimal particle; in the current particle swarm the optimal particles become the group optimal particles. The mathematic descriptions of algorithm are as follows.

Suppose the swarm consisting of M particles searches the n dimensional space (it's the particle's dimension number), the position of each particle represents as $X_i = (x_{i1}, x_{i2}, \ldots, x_{ij}, \ldots, x_{in})$, the fitness function is F which is corresponding to each position and related to the optimization objective function, the corresponding speed of each particle can be expressed as: $V_i = (v_{i1}, v_{i2}, \ldots, v_{ij}, \ldots, v_{in})$. The optimal position in the history flying of single particle is $P_i = (p_{i1}, p_{i2}, \ldots, p_{ij}, \ldots, p_{in})$, which is called the individual extremum pbest particle; the optimal positon of all particles in the group experienced in the history is $P_j = (p_{j1}, p_{j2}, \ldots, p_{cd}, \ldots, p_{jn})$, which is called the group global extremum pbest particle; the updating formula of the i^{th} particle in the k-th iteration with the d-th dimension speed v_{id}^{k+1} and position x_{id}^{k+1} is:

$$v_{id}^{k+1} = wv_{id}^{k} + c_1 r_1 (pbest - x_i^k d) + c_2 r_2 (gbest - x_{id}^{k}) \quad (12)$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \quad (13)$$

In the expressions, w is the inertia weight; c_1 and c_2 are the positive acceleration constants or acceleration factors, respectively represent the weight coefficients for a single particle tracking its history optimal value and the weight coefficients of tracking group optimal. The appropriate c_1 and c_2 can accelerate the convergence speed and be not easy to fall into local optimum; r_1 and r_2 and are the uniformly distributed random numbers between the (0, 1). The position change range of the d^{th} dimension is ($-XMax_d$, $XMax_d$). The speed change range is ($-VMax_d$, $VMax_d$), in the iteration process, if the position and the speed exceed the boundaries, then taking the boundary value. Clerc and Kennedy (2002) analyzed the above parameters and provided the parameters conditions of the PSO algorithm convergence.

The adjustment of inertia weighting factor w: The inertia weighting factor w represents the inertia that particles keep their movement, makes it have search ability, when w takes big value, it is helpful to search a new area; but when w takes a small value, it is helpful for particles to search carefully in the current region. The inertia weight factor in this study adopts the decreasing strategy, that is:

$$w = w_0 - (w_0 - w_1)\frac{N_i}{N_{\max}} \tag{14}$$

In the expression,

N_{max} = The maximum iteration time
N_i = Current iteration time
w_1 = The inertia weight when the particles evolve to the maximum algebra

THE PARTICLE SWARM OPTIMIZATION ALGORITHM

Particle encoding strategy: The key problem of particle swarm optimization algorithm is the particle's position is corresponding to the solution, this study references the thinking of literature (Salmen *et al*., 2002) to construct a new particle encoding method, the 3n dimensional vector X is used to express the farmland food harvester routing optimization problem with m depots[9], l farmland food harvesters and n customer tasks. The particle's first dimension X_r represents the depot information to serve customers; the second dimension X_s represents the farmland food harvester information to serve customers; the particle's third dimension X_t represents the driving distance of each farmland food harvesters.

Particle's decoding strategy: Hereinto, X_t expresses the distance of each farmland food harvester, if the driving path sequence of farmland food harvester i needs converting, X_t must be adjusted. The adjustment function Af can be determined according to the size and the sequence of vector element X_t, that is, at first looking for the farmland food harvester service demand point i and then according to the size of X_t corresponding to i to sort the numbers increasingly, so as to determine the driving path sequence of farmland food harvester i. For example, suppose there are 10 customers and 2 depots and there are 2 or 3 trucks respectively, if a particle's position vector X is shown in Table 1, the adjusted position vector X is shown in Table 2.

Table 1: The position vector X before adjustment

Cus.	1	2	3	4	5	6	7	8	9	10
X_r	1	2	2	1	2	1	2	1	2	2
X_s	1	4	3	2	4	1	4	2	5	5
X_t	0.3	3.2	2.6	1.3	4.3	0.1	4.6	0.6	5.8	5.9

Table 2: The position vector X after adjustment

Cus.	1	2	3	4	5	6	7	8	9	10
X_r	1	2	2	1	2	1	2	1	2	2
X_s	1	4	3	2	4	1	4	2	5	5
X_t	2	1	1	2	2	1	3	1	1	2

Then the corresponding solution paths are as follows:

Depot 1: farmland food harvester l: 6→1
Farmland food harvester 2: 8→4
Depot 2: farmland food harvester 3: 3
Farmland food harvester 4: 2→5→7
Farmland food harvester 5: 9→10

The particle encoding and decoding can ensure that every customer is served and limit each customer can be performed only by one truck, to make the feasible process calculation of solution greatly reduced.

The implementation process of the algorithm:

Step 1: **Initialize the algorithm:** Set the parameters, inertia weight w, acceleration factors c_1 and c_2, determine population size H and the maximum iterations time $iter_{max}$.

Step 2: **Initialize the particle swarm:** The particle swarm space is divided into several adjacent subgroups, which requires both of the two adjacent subgroups overlap with each other.

The position x of each particle randomly takes the integer between (I, n+m-1), each particle velocity vector v randomly takes the real number in the following space (- (n + m - 2), (n + m - 2)).

The integer serialization adjustment function Af is used to adjust all particles in the particle swarm.

The history optimal pbest of individual particles is set as the current position and the position of optimal individual particle in the population is regarded as the current *gbest*.

Step 3: Repeat the following steps, until reach the largest iterations times:

- According to expression (14) to update the current inertia factors.
- According to expression (12) and (13) to update the position and velocity of each particle in the particle swarm, when *x* and *v* succeed their scope, take the values according to the boundary values.
- Utilize the integer serialization adjustment function *Af* to adjust all particles in the particle swarm.
- If the current function value of a particle is superior to its individual extreme value *pbest*, the current position of the particle is taken as its individual extreme value *pbest*; if it is superior to the swarm extremum *gbest* in particle group, the current particle position is used to replace group extreme value *gbest*.
- Search the current optimal individual and optimal group within the adjacent particle swarms, if it is superior to the history optimal solution, then update the individual optimal *pbest* and group extreme *gbest* inside particle swarm.

Step 4: Output the current optimal solution.

ANALYSIS OF EXAMPLES

Schedule the farmland food harvester routing containing 3 depots and 25 customer points. Customers and depots location are distributed in the area with 10×10 (unit: km), the farmland food harvester driving speed is 25 km/h, customer location (unit: km), demand (unit: t) and the latest time (unit: min) allowed to arrive are shown in Table 3, the depots location and the number of farmland food harvesters are shown in Table 4, the maximum load of each farmland food harvester is 20 tons/truck.

Respectively genetic algorithm, ant colony algorithm and particle swarm optimization algorithm are used to do calculation of this example on the same computer. The optimal scheduling scheme to solve this example is seven trucks from three depots joining in the scheduling, the driving route of each truck, driving distance (unit: km) and driving time (unit: min) are shown in Table 5.

It is not hard to see from the calculated results of the example, in this study the adopted algorithm can

Table 3: Customer information

Serial number	X coordinates/km	Y coordinates/km	Demand/t	Delay time/min
1	4.3	8.2	7	40
2	3.4	1.1	6	10
3	9.5	8.3	9	60
4	2.2	5.3	5	10
5	7.3	4.1	7	10
6	5.1	4.8	4	30
7	5.9	6.8	5	15
8	4.0	0.2	7	40
9	1.8	9.9	6	20
10	6.7	2.0	3	20
11	0.6	2.7	3	30
12	7.8	10.0	4	40
13	3.9	5.1	7	10
14	0.3	7.7	6	20
15	6.0	0.9	2	30
16	1.1	0.9	4	15
17	9.5	5.2	8	50
18	0.4	6.1	3	20
19	1.9	4.7	5	60
20	6.3	2.6	8	10
21	8.8	1.6	3	20
22	2.0	7.9	5	50
23	6.2	3.1	8	40
24	0.5	2.3	1	20
25	1.2	7.0	4	30

Table 4: The information of depots

Depot	26	27	28
Location	(3, 7)	(4, 3)	(8, 6)
Owned farmland food harvesters	5	5	5

Table 5: The optimal scheduling plan of the example

Depots	Driving route	Driving distance/km	Driving time/min
26	26→4→18→25→22→26	7.60	22.80
26	26→→14→9→1→26	10.24	30.72
27	27→2→16→24→11→19→27	11.32	33.96
27	27→20→10→15→8→27	9.28	27.84
28	28→5→21→17→28	10.31	30.93
28	28→7→12→3→28	11.12	33.36
Summation		67.44	202.32

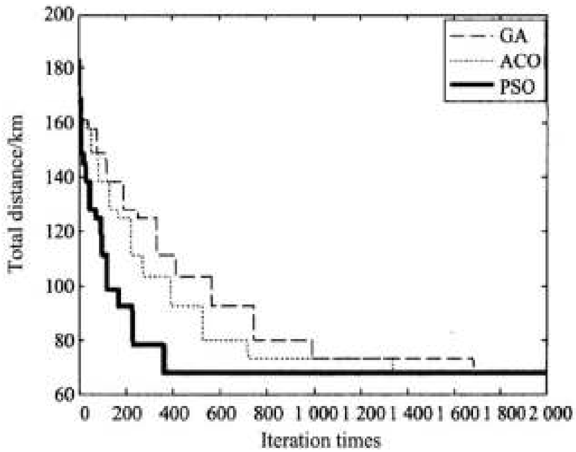

Fig. 1: The GA, ACO and PSO performance comparisons of the example

quickly find out the optimal solution of multiple-depot farmland food harvester routing problem with time windows and the time and efficiency to search the optimal value are better than genetic algorithm and ant colony algorithm, which provides a new method for solving multiple-depot farmland food harvester routing problem with time windows. The performance comparisons of the example among three algorithms are shown in Fig. 1, which also prove this result.

CONCLUSION

According to the actual situation of logistics distribution, this paper studies a kind of more complex multiple-depot farmland food harvester routing problem with time windows and aiming to the multiple-depot farmland food harvester routing problem to construct a particle swarm coding method, establishes the corresponding mathematical model and solving algorithm, through the example test, in a very short period of time the optimal solution of multiple-depot farmland food harvester scheduling is solved, which has strong applicability to logistics distribution.

ACKNOWLEDGMENT

This study is supported by the funds of the science and technology research key project of Henan province (2014), China. The project name is “The logistics distribution farmland food harvester scheduling model and its optimization based on the dynamic demand” and the project number is 142102210231.

REFERENCES

Chaos, I.M., B.L. Golden and E. Wasil, 1983. A new heuristic for the multi-depot vehicle routing problem that improves upon best-known solutions. Am. J. Math. Manage. Sci., 13: 371-406.

Clerc, M. and J. Kennedy, 2002. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE T. Evolut. Comput., 6: 58-73.

Dantzig, G. and J. Ramser, 1959. The truck dispatching problem. Manage. Sci., 6: 58-102.

Kennedy, J. and R. Eberhart, 1995. Particle swarm optimization. Proceeding of the IEEE International Conference on Neural Networks, pp: 1942-1948.

Meijun, C., Z. Zhisheng and C. Chunyong, 2008. A new type of ant colony clustering algorithm of multiple-depot vehicle routing problem. China's Manuf. Ind. Inform., 37: 1-5.

Salmen, A., I. Ahmad and S. Al-Madani, 2002. Particle swarm optimization for task assignment problem. Microprocess. Microsy., 26: 363-371.

Yuanbiao, Z. and L. Guangqing, 2007. The logistics distribution route optimization problem study based on hybrid particle swarm optimization. J. Packag. Eng., 28: 10-12.

Yuanfeng, Y., 2008. The improved genetic algorithm for multiple-depot multiple-vehicles vehicle routing problem. Comput. Moderniz., 9: 10-13.

Zhixiong, L., 2009. Logistics vehicle optimize scheduling research based on particle swarm algorithm. J. Wuhan Univ., Sci. Technol., 32: 615-618.