A PARTICLE SWARM OPTIMIZATION ALGORITHM FOR MULTI-DEPOT CAPACITATED LOCATION-ROUTING PROBLEM WITH INVENTORY DECISIONS IN SUPPLY CHAIN NETWORK DESIGN

Mahdi Bashiri[1*] and Ehsan Fallahzade[2]
[1]Department of Industrial Engineering
Shahed University, Tehran, Iran
bashiri@shahed.ac.ir

[2] Department of Industrial Engineering
Shahed University, Tehran, Iran
e.fallahzade@shahed.ac.ir

ABSTRACT

Location routing problem is one of main problems in location analysis which contains both strategic and tactical decisions. This problem can be more applicable when the inventory policies are investigated. In this paper we present a particle swarm optimization algorithm to solve a model which considers location, allocation, capacity, inventory and routing decisions in a stochastic supply chain network. Each depot keeps certain amount of safety stock to reduce the risk of uncertainty. This uncertainty comes from customer demands that follow a normal distribution. The proposed solution method optimizes the location, routing and inventory problems simultaneously. The specific feature of the proposed algorithm is considering the location and routing problems together in a single stage when searching in the feasible space to find the best solution. The proposed approach was analyzed by some simulated numerical examples and the results compared by an exact solution approach. The results show that the proposed solution approach performs more efficiently.

[*] Corresponding Author

1 INTRODUCTION:

One of the most important segments of a supply chain network is its distribution centers. There are few articles that simultaneously consider location, allocation, capacity, inventory and routing decision together in this field. Min [14], and Nagy, G., Salhi, S [15] had surveyed location-routing problems and presented classifications on these problems. Inventory–routing problems were studied in several papers, e.g. Baita [4], Jaillet [10], Kleywegt [13], Adelman [1], Gaur [9], Zhao [23], Yu [22], Oppen J. [16], and Day J.M. [7]. Also, Erlebacher [8], Daskin M. [6], and Shen Z. [18] have been studied location–inventory problems.

Recently few articles such as Shen Z. Q. [19], and Javid A., Azad N. [11], considered all mentioned fields together in their studies.

Perl J., Daskin M.S. [17] Showed that location-routing problems are in the class of NP-Hard problems, consequently the problems that additionally consider inventory decisions belong to NP-hard problems too. Because of complexity, the instances with a large number of customers, distribution centers or vehicles cannot be solved within acceptable time. Therefore, many heuristic and meta-heuristic algorithms have been developed in order to find optimal or near optimal solutions in reasonable computational time.

Some authors like Yang et al. [21] believed that PSO has some properties which make it easy to implement with a tuned parameters. Some other researchers Chen A., et. al. [5]; Tao et. al. [20] claim that PSO can perform more efficiently when hybridized with local search. Kennedy and Eberhart [12] are believed to be the pioneers of the PSO concept which is a kind of swarm intelligent algorithm based on socio-psychological principles. It has been applied to several routing problems with success in other occasions. For example Ai, J., & Kachitvichyanukul [2] developed a PSO for a vehicle routing problem (VRP) with simultaneous pick-up and delivery, and compared its performance with other existing meta-heuristics. They used a similar PSO for the capacitated VRP (CVRP) and they reported some promising results at Ai J., & Kachitvichyanukul [3].

Javid A., Azad N. [11] used a hybrid tabu search (TS) and simulated annealing (SA) approach to solve their model which considers location, routing and inventory together. The algorithm that they used is a two stage algorithm that after constructing initial solution, in the first stage, it tries to improve location problem, the output of this stage will be input of second stage which tries to improve routing problem.

In this paper we present a particle swarm optimization algorithm in which both location and routing problems are considered simultaneously, and the results will be compared to exact solutions.

2 PROBLEM DESCRIPTION AND FORMULATION

The purpose of the model includes selecting and locating depots (distribution centers) which are chosen from a set of potential ones, selecting a capacity level for each depot, allocating customers to each selected depot, specifying the inventory policy and finally scheduling vehicles' routes to meet customers' demands with a minimum total cost. The model assumes that each customer's demand follows a normal distribution. As mentioned before the model has been extracted from Javid A., Azad N. [11].

2.1 Index sets

K set of customers

J set of potential distribution centers

M merged set of customers and potential distribution centers, i.e. $(K \cup J)$

N_j set of capacity levels available to distribution center $(j \in J)$

V set of vehicles

2.2 Parameters and notations

μ_k Mean of yearly demand at customer k ($k \in K$)

σ_k^2 Variance of yearly demand at customer k ($k \in K$)

f_j^n Yearly fixed cost for opening and operating distribution center j with capacity level n ($\forall j \in J, \forall n \in N$)

d_{kl} Transportation cost between node k and node l $\forall k, l \in M$

q Number of visits of each vehicle in a year

h_j inventory holding cost per unit of product per year at distribution center j ($\forall j \in J$)

p_j fixed cost per order placed to the supplier by distribution center j ($j \in J$)

lt_j lead time of distribution center j in years ($j \in J$)

g_j fixed cost per shipment from supplier to distribution center j ($j \in J$)

a_j cost per unit of shipment from the supplier to distribution center j ($j \in J$)

α Desired percentage of customer orders that should be satisfied (fill rate), $\alpha > 0.5$

z_α Left α-percentile of standard normal random variable Z, i.e. $P(Z \le z_\alpha) = \alpha$

β Weight factor associated with transportation cost

θ Weight factor associated with inventory cost

$$R_{klv} = \begin{cases} 1 & \text{if k precedes l in route of vehicle v} \\ 0 & \text{otherwise} \end{cases}$$

$$Y_{jk} = \begin{cases} 1 & \text{if customer k is assigned to distribution center j} \\ 0 & \text{otherwise} \end{cases}$$

$$U_j^n = \begin{cases} 1 & \text{if distribution center j is opened with capacity level n} \\ 0 & \text{otherwise} \end{cases}$$

The following function is total cost and we are going to minimize it:

$$\sum_{j\in J}\sum_{n\in N_j} f_j^n U_j^n + \beta q \sum_{v\in V}\sum_{k\in M}\sum_{l\in M} d_{kl} R_{klv} + \sum_{j\in J}\left[\sqrt{2\theta h_j(\theta p_j + \beta g_j)\sum_{k\in K}\mu_k Y_{jk}} + \beta a_j \sum_{k\in K}\mu_k Y_{jk} + \theta h_j z_\alpha \sqrt{lt_j \sum_{k\in K}\sigma_k^2 Y_{jk}}\right] \quad (1)$$

The first term of the objective function is the fixed cost of locating the open distribution center, the second term is costs associated to routing problem and last term represents inventory costs.

3 PROPOSED PSO ALGORITHM

PSO uses set of initial solution called particles, each particle moves through space according to following vectors:

- Continue their current moving vector
- A local best position which is the best that the particle has experienced (p_best).
- A global best position which is the best position found by all particles till now (g_best).

The moving speed toward each particle will be considered by certain coefficients.

Considering location and routing decisions together increases the randomization structure of the algorithm and it will help us to escape from trapping in local optimum.

The parameters of the PSO algorithm are as following:

Table 1 Proposed PSO algorithm's parameters

Parameters	**Definition**
Iteration	Number of outer loop repeat
Number of particles	Number of initial solutions
B1	Coefficient related to the speed of moving toward personal best
B2	Coefficient related to the speed of moving toward global best

3.1 Solution representation

The representation that we use for our solution is consists of two row vectors:

- The first row defines which distribution center at which capacity level is selected, which customers has been allocated to each distribution center; and also indicates the order, in which the customers are being serviced.
- The second row shows the routs and the number of vehicles that we need to satisfy the customers' demands.

To illustrate consider the following solution:

20	5	3	2	5	12	6	4	1	6
0	0	1	2	0	0	0	3	3	0

Figure 1-Proposed solution representation

In the first row each cell immediate before first depot is the selected capacity of the corresponding depot. The numbers between two depot are the customers that has been allocated to that depot, the order of these numbers is also important because it shows the sequence in which customer has been serviced. So here the distribution center 5 is opened with the capacity of 20 units and the customers 3 and 2 are allocated to this depot. Likewise depot 6 is opened at the capacity level of 12 and customers 4 and 1are allocated to it.

According to second row vector in this example we have 3 vehicles so as a result we will have 3 different routes, customer 3 is getting service by vehicle number 1 that starts it's rout from depot 5, customer 2 is getting serviced by vehicle 2, same as vehicle 1, it starts it's rout from depot 5. Vehicle 3 is going to service customers 4 and 1, starting its trip from depot 6 go to customer 4 then 1 and finally return to depot 6. Figure1 illustrates aforementioned discussion clearly.

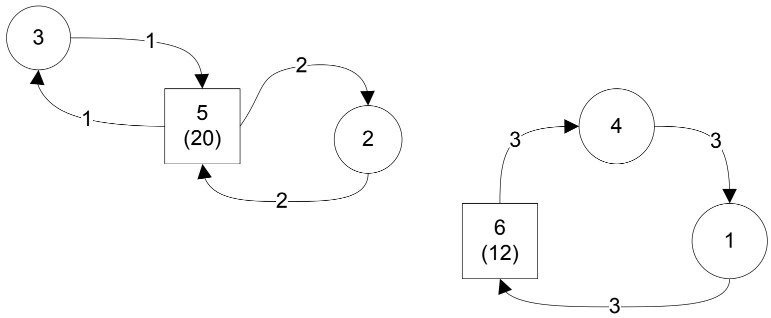

Figure 2- Schematic interpretation of the example representation

3.2 Generating particles

In order to build initial particles, consider set k' and d' , the element of these sets are the non-repeated permutation of the customers and potential depots sets, respectively. The following algorithm shows how the particles are constructed as initial feasible solutions, using mentioned sets. (Repeat the following algorithm until it builds all needed feasible particles)

- ***Step1:*** Select the first element of d' , and randomly choose a capacity from available capacity levels for this element. If the selected distribution center is the last element of d' go to *Step3* otherwise go to *Step2*.
- ***Step2:*** Starting from first element of k' randomly select *n* of them, one by one allocate these to the distribution center selected from Step1 until it exceeds the selected capacity level, if so close the distribution center, delete it from d' and remove the allocated customers from set k' . Simultaneously allocate vehicles to the customers, if the capacity of vehicle is violated use the next vehicle. Return to *Step1*.
- ***Step3:*** In this step all the remaining customers must be allocated. Start from first element of k' , one by one allocate customers to the distribution center selected from Step1 until it exceeds the selected capacity level, and if so, choose larger capacity level for this distribution center. If it is impossible to allocate all remaining customers to the last distribution center, reset k' and d' then go to *Step1*.

3.3 Moving toward personal and global best

In order to make particles move toward personal (global) best we perform the following procedure:
We consider the sequence of customers and depots in the personal (global) best particle, then we *b1* (*b2*) times change the sequence of each particle's customers and depots, in a way that it becomes like personal (global) best sequence, after that the particles will be rebuilt by new sequence of customers and distribution centers.

3.4 The proposed PSO Algorithm

Step1: Initialize k particles by the algorithm proposed in section 3.2, and set
zp_best=infinite
zg_best=infinite

For *iter=1,…, iteration do the following steps:*

Step2: For each particle randomly change two customers' positions and calculate the corresponding total cost according to equation 1.
If the calculated total cost is smaller than the *zp_best set:*
zp_best=total cost;
ap_best=current solution

Step3: Find the minimum total cost among all particles and set:
zg_best=min (total costs)
And put the corresponding solution into *ag_best*.
Step4: According to the procedure proposed in section 3.3 move toward personal and global best.

As mentioned before the benefit of proposed algorithm is considering the location, allocation and routing decisions together. This consideration will expand the search space.

4 COMPUTATIONAL RESULTS

In order to verify efficiency of the proposed algorithm, several simulated instances are used. The model and the exact solution are implemented in GAMS program, and the proposed PSO algorithm is coded in MATLAB 7.12.0 (R2011) on a PC with 1.73 GHz INTEL Dual Core CPU and 2GB RAM. Table 1 consists of 25 different scenarios and their corresponding CPU times and objective functions. Each objective and CPU time value is the average of 5 independent runs.

Table 2- Simulated numerical examples

No.	# of depots	# Of customers	Vehicle capacity	Maximum available vehicle	# of used vehicles	Exact Time	PSO Time	Exact objective	PSO objective
1	2	3	10	3	3	3.24	1.7	130.625	130.625
2	2	3	50	2	1	4.34	1.9	123.725	123.725
3	2	4	15	2	2	6.13	2.4	129.135	129.135
4	2	3	10	4	3	11.65	2.53	130.625	130.625
5	2	3	15	2	2	14.39	2.56	127.325	127.325
6	3	3	15	2	2	20.45	2.8	110.206	110.206
7	3	3	10	3	3	32.88	2.801	112.606	112.606
8	3	3	50	2	1	37.83	2.81	107.806	107.806
9	2	4	50	2	1	68	2.99	125.835	125.835
10	3	4	15	2	2	74	3.01	128.197	125.835
11	3	4	15	2	2	77	3.15	128.1973	128.197
12	3	3	10	4	3	83.05	3.159	112.606	112.606
13	3	4	50	2	1	85	3.1611	125.835	125.955
14	3	4	10	3	3	140	3.32	129.654	129.680
15	2	4	10	3	3	156.26	3.3388	129.583	131.835
16	4	4	15	2	2	176.23	3.36	120.207	120.567

No.	# of depots	# Of customers	Vehicle capacity	Maximum available vehicle	# of used vehicles	Exact Time	PSO Time	Exact objective	PSO objective
17	4	4	50	2	1	193.42	3.37	119.607	119.787
18	2	4	10	4	3	325.2	3.7	131.835	131.835
19	4	5	15	3	3	494.13	3.8	132.19	132.433
20	3	4	10	4	3	1000	3.8	129.654	129.662
21	3	4	15	3	2	1000	3.8	128.197	125.895
22	4	4	10	4	3	1000.14	3.8	183.388	121.407
23	4	4	10	3	3	1000.2	3.86	121.407	121.407
24	4	5	10	4	4	1000.39	4.3	161.303	134.413
25	4	5	50	2	2	1000.42	4.7	128.89	128.89

Figure 3 verifies that even in small instances, the proposed algorithm causes a significant time reduction in solving the problem. Figure4 depicts objective function values obtained from both methods. We can see that they are relatively similar, even in some cases the proposed algorithm achieves better objective value than the exact solution approach.

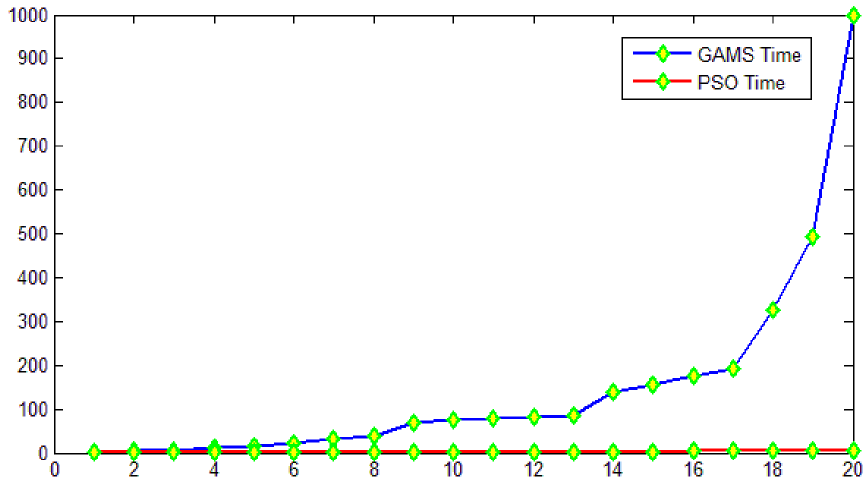

Figure 3 - Time comparison between exact and proposed PSO solution approaches

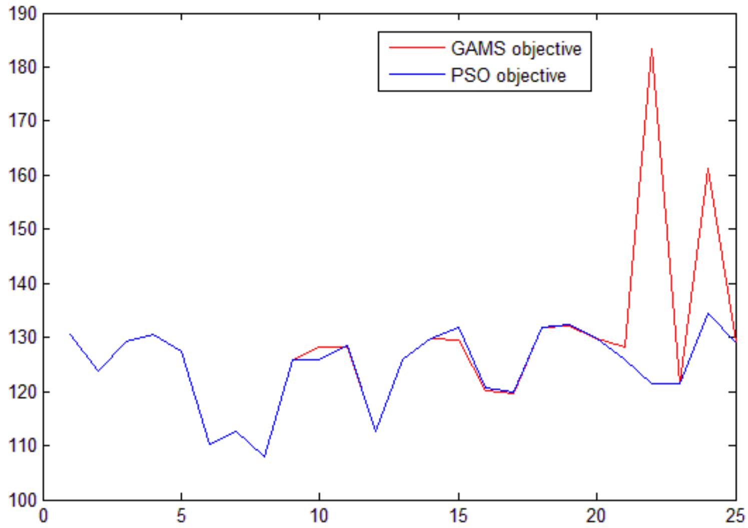

Figure 4- Objective values comparison between exact and proposed PSO solution approaches

5 CONCLUSION

Particle swarm optimization is one of the most effective meta-heuristic optimization methods. We use this effective algorithm to solve one of the practical models in supply chain network design, which incorporate inventory, location allocation and routing problems. The specific feature of the proposed algorithm is that it considers the location and routing problems together when searching in the feasible space to find the best solution. Numerical examples also declared that the proposed algorithm performs more effectively. We can see that our proposed method can solve the problem in a significant less time.

Further extensions could be modifying the solution representation of the proposed PSO algorithm, also adding a neighbourhood search to the algorithm could increase its efficiency.

6 REFERENCES

[1] **Adelman, D. (2004).** A price-directed approach to stochastic inventory/routing. *Operations Research 52* , 499-514.

[2] **Ai J., & Kachitvichyanukul V. (2009b).** Particle swarm optimization and two solution representations for solving the capacitated vehicle routing problem. *Computers & Industrial Engineering, 56* , 380–387.

[3] **Ai, J., & Kachitvichyanukul, V. (2009a).** A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. *Computers & Operations Research 36* , 1693–1702.

[4] **Baita, F. U. (1998).** Dynamic routing-and-inventory problem: a review. *Transportation Research: Part A 32* , 585-598.

[5] **Chen A., Yang G., & Wu Z. (2006).** Hybrid discrete particle swarm optimization. *Journal of Zhejiang University* , 607-614.

[6] **Daskin M., C. C. (2002).** An inventory-location model: formulation, solution algorithm and computational results. *Annals of Operations Research (2002)* , 83-106.

[7] **Day J.M., W. P. (2009).** Improving routing and scheduling decisions at a distributor of industrial. *Omega 37* , 227-237.

[8] **Erlebacher, S. M. (200).** The interaction of location and inventory in designing distribution systems. *IIE Transactions 32* , 155-166.

[9] **Gaur, V. F. (2004).** A periodic inventory routing problem at a supermarket chain. *Operations Research 52* , 813-822.

[10] **Jaillet, P. B. (2002).** Delivery cost approximations for inventory routing problems in a rolling horizon framework. *TransportationScience 3* , 292-300.

[11] **Javid A., Azad N. (2010).** Incorporating location, routing and inventory decisions in supply chain network design. *Transportation Research Part E 46* , 582-597.

[12] **Kennedy, J., & Eberhart, R. (1995).** Particle swarm optimization. *IEEE international connference neural networks*, (pp. 1942-1948). Perth, Australia.

[13] **Kleywegt, A. N. (2002).** The stochastic inventory routing problem with direct deliveries. *Transportation Science 36* , 94-118.

[14] **Min, H. J. (1998).** Combined location-routing problems: a synthesis and future research directions. *European Journal of Operational Research 108* , 1-15.

[15] **Nagy, G., Salhi, S. (2007).** Location-routing, issues, models, and methods: A review. *European Journal of Operational Research 117* , 649-672.

[16] **Oppen J., L. A. (2008).** A Tabu search approach for the livestock collection problem. *Computers and Operations Research 35* , 3213-3229.

[17] **Perl J., Daskin M.S. (1985).** A warehouse location-routing problem. *Transportation Research Part B 19* , 381-396.

[18] **Shen, Z. (2005).** A multi-commodity supply chain design problem. *IIE Transactions 37* , 753-762.

[19] **Shen, Z. Q. (2007).** Incorporating inventory and routing cost in strategic location models. *European Journal of Operational Research 179* , 372-389.

[20] **Tao et. al. (2008).** A Mixed PSO algorithm for the VRPSPD. *In Control and decision conference, May, Chinese* , 4017-4021.

[21] **Yang S. Y., Wang, M., & Jiao L. C. (2004).** A quantum particle swarm optimization. *Congress on evolutionary computation, June, Portland, USA* , 320-324.

[22] **Yu, Y. C. (2008).** A new model and hybrid approach for large scale inventory routing problems. *European Journal of Operational Research 189* , 1022-1040.

[23] **Zhao, Q. C. (2008).** Model and algorithm for inventory/routing decision in a three-echelon logistics system. *European Journal of Operational Research 191* , 623-635.