Int J Adv Manuf Technol (2016) 85:2345–2352
DOI 10.1007/s00170-015-8081-3

ORIGINAL ARTICLE

A hybrid heuristic method for the periodic inventory routing problem

Shu-Chu Liu [1] · Ming-Che Lu [1] · Chih-Hung Chung [2]

Received: 15 May 2015 / Accepted: 5 November 2015 / Published online: 17 November 2015

Abstract The periodic inventory routing problem (PIRP) determines the delivery routing and the inventory policies for retailers from a supplier in a periodic time based on the minimal cost criterion. Since it is a non-deterministic polynomial-time (NP)-hard problem, a heuristic method is needed for this problem. In the past, different global heuristic methods, such as tabu search (TS) and simulated annealing (SA), have been proposed; however, they seem ineffective. Particle swarm optimization (PSO) is known as resolving multidimensional combinatorial problems such as PIRP; however, it is easily trapped in local optimality. The authors of this paper propose a hybrid heuristic method for the PIRP. The hybrid method integrates a large neighborhood search (LNS) into PSO to overcome the drawbacks of PSO and LNS. The PSO is adopted first. A local search is applied to each particle in different iterations. Then, a local optimal solution (particle) for each particle is obtained. Last, the LNS is applied to the global best solution to avoid becoming trapped in local optimality. The results show that the proposed hybrid heuristic method is 10.93 % better than the existing method and 1.86 % better than the pure heuristic method in terms of average cost.

Keywords Periodic inventory routing problem (PIRP) · Particle swarm optimization (PSO) · Large neighborhood search (LNS) · Hybrid heuristic method · NP-hard problem

✉ Chih-Hung Chung
chih-hungchung@my.unt.edu

Shu-Chu Liu
sliu@mail.npust.edu.tw

1 Department of Management Information Systems, National Pingtung University of Science and Technology, Pingtung, Taiwan

2 Department of Learning Technology, National Pingtung University of Science and Technology, Pingtung, Taiwan

1 Introduction

The inventory routing problem (IRP) has been discussed and adopted in different industries [1, 20]. When supply chain management (SCM) and vendor-managed inventory (VMI) are adopted, they become strategic tools for companies to save money. In the last three decades, companies have seen an increasing importance placed on research in IRP. Since the first integrated IRP proposed by Federgruen and Zipkin [4], IRP has shown considerable efficiency. The IRP system could save approximately 40 % of the total working hours in large soft drink firms [19]. Moreover, studies by Gaur and Fisher [6] and Fu and Fu [5] proposed that an IRP system could decrease an organization's total cost and improve their advantage, compared to the approach of separating optimization.

Recently, the periodic inventory routing problem (PIRP) has been garnering considerable attention, not only because of the way it is applied, but also because it is from scientific research [18, 21]. The PIRP determines vehicle routing and delivery times for retailers from a supplier in a repeated period based on the minimal transportation and inventory cost criterion. It occurs often in practice, such as Albert Heign, a leading supermarket chain in the Netherlands [6], Walmart, and others [21]. In the USA, car manufacturers need to set up a periodic time to deliver new cars and accessories to car dealerships because of the long distance. Therefore, PIRP is an important topic in both practical and academic implications.

Comparing IRP and PIRP, IRP helps to allocate inventory and decide routing schedules simultaneously in a supply chain system. In order to achieve a global solution, IRP minimizes the total cost (the distribution and inventory costs of retailers) [21]. PIRP is a multidimensional problem that includes delivery times for retailers and vehicle routing for retailers in a fixed and repeated period. Since searching for an optimal solution for a multidimensional problem is a non-deterministic polynomial-time (NP)-hard problem, it is not feasible to adopt a mathematical method to resolve the problem [16]. Hence, a heuristic method is needed. In the past, different heuristic methods, such as a tabu search (TS) [21], simulated annealing (SA) [29], etc., have been proposed for the PIRP. Because the structure of IRP is similar to that of PIRP, the IRP is also reviewed.

Since the neighborhood-based approach can search different dimensions quickly and can be addressed easily [8, 9], it is often adopted for the IRP. The related approach includes a local search [4, 7], TS [13, 28, 30], SA [29], and a variable neighborhood search (VNS) [15, 23, 31]. In addition, a VNS is better than a TS, SA, and local search [27, 31]. However, the VNS lacks a learning mechanism. Recently, the particle swarm optimization (PSO), an approach based on an evolving population, has become more feasible for a multidimensional problem such as PIRP [2, 11]. Furthermore, the PSO has been found to be superior to neighborhood-based approaches [17, 24], although it is easily trapped in local optimality. In addition, since PIRP needs more complex representation and processes to search for the global optimal solutions, other population-evolving-based approaches such as ant colony optimization (ACO), path relinking, scatter search, and parallel computing approaches such as deoxyribonucleic acid computing are inappropriate for solving the PIRP [25]. Hybrid heuristic methods for multidimensional problems have proven to be better than a single pure heuristic method [3]. For example, Küçükoğlu and Öztürk [12]) proposed a hybrid method integrating TS and SA to overcome the shortcomings of using pure heuristic methods. Li et al. [14] proposed a hybrid method integrating TS and VNS, and it proved to be better than TS, SA, and GA. Marinakis et al. [17] proposed a hybrid method integrating PSO and VNS, and it proved to be better than PSO. However, according to the literature review, VNS outperforms SA, TS, and GA in vehicle routing problems (VRP) [27, 31]. Compared to VRP, PIRP needs more complex processes to search the global optimal solutions. Additionally, the ACO approach presented lower performance for resolving PIRP compared to other hybrid algorithms such as TS, GA, VNS, and SA (Cho, Lee, Lee, & Gen, 2014; Dehbari, Pour Rosta, Ebrahim Nezhad, Tavakkoli-Moghaddam, & Javanshir, 2012; [25]. Wang et al. [26] proposed a hybrid method integrating PSO and GA, and it proved to be better than PSO and GA. According to the literature review, a hybrid heuristic method integrating PSO and large neighborhood search (LNS) to overcome the shortcomings of using pure heuristic methods is best (LNS is a transformation of VNS). Therefore, it makes more sense to adopt the hybrid heuristic method for solving the PIRP in this paper.

2 The proposed hybrid heuristic method

Before the proposed hybrid heuristic method is introduced, the authors should mention the formulation for the model first. The model formulation for PIRP is the same as that in Qin et al. [21] (please refer to Appendix for the details). Since finding the optimal solution for the PIRP is an NP-hard problem, a heuristic method is adopted. A hybrid heuristic method integrating PSO and LNS is proposed to overcome the shortcomings of PSO or LNS in this paper. The initial solutions (particles) and their corresponding velocities are generated randomly by the algorithm. Consequently, the algorithm will decide the global best solution and particle best solution. A local search is applied to each particle in different iterations in order to find better solutions around each particle (since the solutions [particles] found by PSO can be in a random position and better neighborhood solutions are ignored, a local search finds better neighborhood solutions of PSO-generated solutions). The local search is executed repeatedly until no better neighborhood solution is found. Then, a local optimal solution (particle) is obtained, and the global best solution and particle best solution are revised. The LNS is applied to the global best solution to avoid becoming trapped in local optimality. If the new solution is better than the global best solution, the global best solution is substituted with the new solution. The position and velocity for each particle in the new iteration are then revised. After a specific number of iterations are executed, the final global best solution of the proposed hybrid heuristic method is found, and the procedure stops (Fig. 1).

2.1 Initialize parameters

Set $ite=1$ (the index of current iteration), $ind=1$ (the index of current particle), I (the maximal iteration), np (the number of particles), and $nlns$ (the iteration for LNS).

2.2 Generate the position and velocity for each particle in the first iteration

There are np particles generated randomly in the first iteration. Each particle X includes period information, vehicle information, routing information, and threshold information. The representation for X (solution) is shown in Table 1. There are six retailers (from A to F) served by three vehicles (from V1 to V3) in two periods (from T1 to T2). The value is randomly generated from U[0, 10] (the range is experimentally decided based on the minimal cost criterion). The threshold values are 5.1 for the first period and 7.8 for the second period. The value is randomly generated from U[0, 10]. After Table 1 is generated, it is translated into Table 2 according to the threshold constraint. If the value is less than the threshold value, it becomes 0; otherwise, the value is unchanged.

If the retailer is supplied by different vehicles, the retailer is supplied by the vehicle with the maximal value since each retailer can be supplied only by a specific vehicle in any period (it is experimentally decided based on the minimal cost criterion). If all demands for retailers in a specific vehicle are greater than the vehicle capacity, then the transportation service for the retailers with smaller value is canceled until the vehicle capacity

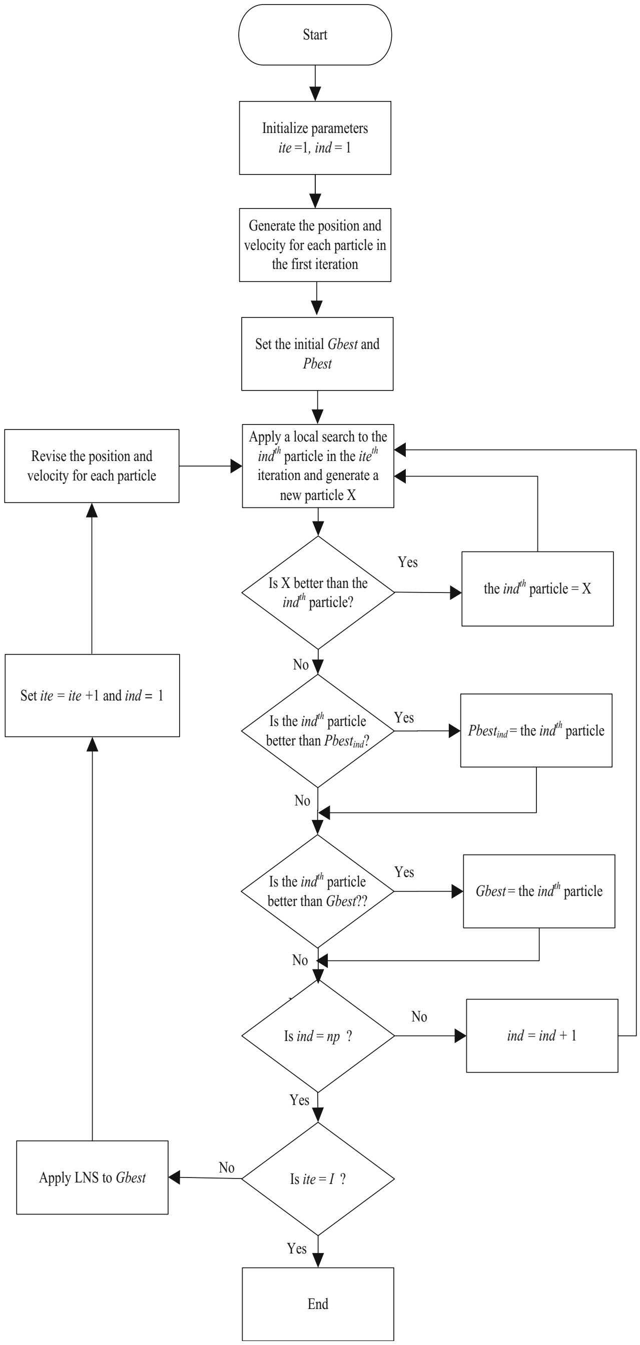

Fig. 1 The flowchart for the proposed hybrid heuristic method

constraint is not violated (it is experimentally decided based on the minimal cost criterion). Retailers B (V1 and V2), C (V1 and V3), D (V2 and V3), E (V1 and V2), and F (V1 and V2) are supplied by different vehicles in T1. According to the above policy, Table 2 is translated into Table 3.

For each vehicle in any period, the retailer with the lower value means that their delivery has a higher priority (it is experimentally decided based on the minimal cost criterion). Based on the policy, Table 3 is translated into Table 4.

Table 1 The initial information for a particle X

T1	A	B	C	D	E	F
V1	4.1	5.7	6.8	3.2	5.4	7.3
V2	3.2	6.3	2.5	8.9	8.4	6.2
V3	5.2	4.5	5.8	8.2	4.1	3.2
T2	A	B	C	D	E	F
V1	4.3	9.2	4.2	3.3	3.1	6.1
V2	2.1	3.2	6.4	6.1	7.5	6.2
V3	8.2	5.3	8.4	7.2	5.4	7.1
Threshold	T1	T2				
Value	5.1	7.8				

After the particle is determined, the objective function value for a specific particle can be computed based on the objective function in the model mentioned in Appendix.

V_{ind}^{ite} is generated randomly from $U[-V_{\max}, V_{\max}]$. $V_{\max}$ is generated using 15 % of each variable range [10].

2.3 Set the initial *Gbest* and *Pbest*

$Gbest = \mathrm{Min}\{obj\,(X_1^1)\,, obj\,(X_2^1)\,, \ldots, obj\,(X_{np}^1)\,\}$ $obj\,(X_{ind}^1)$ is the objective function mentioned in Appendix A for X_{ind}^1 , $1 \leq ind \leq np$). $Pbest_{ind} = X_{ind}^1$ $(1 \leq ind \leq \mathrm{np})$.

2.4 Apply a local search to the *ind^th* particle in the *ite^th* iteration and generate a new particle *X*

For the ind^{th} particle X_{ind}^{ite} in the ite^{th} iteration, a local search is randomly selected and applied to the particle. There are four local search approaches, including 1–0 insertion for routing (randomly selects one retailer from one route and inserts it into the same route or other routes), 1–1 exchange for routing (randomly selects two retailers and exchanges them), replenishment deletion (randomly selects a replenishment point for any specific retailer and deletes the replenishment; the demand is sent by previous delivery), and replenishment addition (randomly selects any specific retailer without replenishment and inserts the

Table 2 The translation under the threshold constraint

T1	A	B	C	D	E	F
V1	0	5.7	6.8	0	5.4	7.3
V2	0	6.3	0	8.9	8.4	6.2
V3	5.2	0	5.8	8.2	0	0
T2	A	B	C	D	E	F
V1	0	9.2	0	0	0	0
V2	0	0	0	0	7.5	0
V3	8.2	0	8.4	7.2	0	0
Threshold	T1	T2				
Value	5.1	7.8				

Table 3 The translation under the constraint of each retailer supplied by one specific vehicle in any period

T1	A	B	C	D	E	F
V1	0	0	6.8	0	0	7.3
V2	0	6.3	0	8.9	8.4	0
V3	5.2	0	0	0	0	0
T2	A	B	C	D	E	F
V1	0	9.2	0	0	0	0
V2	0	0	0	0	7.5	0
V3	8.2	0	8.4	7.2	0	0
Threshold	T1	T2				
Value	5.1	7.8				

replenishment; the demand is sent to satisfy the period needs until the next replenishment). One approach is randomly selected from four and applied to X_{ind}^{ite} in order to generate a new particle X. If the new particle X is better than the original particle X_{ind}^{ite}, X_{ind}^{ite} is substituted by $X(X_{ind}^{ite}=X)$. Otherwise, X_{ind}^{ite} is not changed.

2.5 Is the stop criterion reached?

If $ite=I$, the procedure stops; otherwise, (1) apply LNS to *Gbest* (please refer to *Apply large neighborhood search* (*LNS*) *to Gbest* for the details), (2) set $ite=ite+1$ and $ind=1$, and (3) revise the position and velocity for each particle.

2.6 Apply LNS to *Gbest*

For the purposes of this paper, the LNS method is to integrate four local neighborhood search approaches mentioned above into a heuristic search. LNS applies the four local search approaches with random sequence to *Gbest* and generates a new particle X (after experiments, the search adopting four local search approaches with random sequence is better than that with fixed sequence in terms of average cost). If the particle X is better than *Gbest*, then $Gbest=X$. The iteration number for LNS is *nlns* (LNS is executed *nlns* times).

Table 4 The final representation for X

$T=1$	A	B	C	D	E	F
V1	0	0	1	0	0	2
V2	0	1	0	3	2	0
V3	1	0	0	0	0	0
$T=2$	A	B	C	D	E	F
V1	0	1	0	0	0	0
V2	0	0	0	0	1	0
V3	2	0	3	1	0	0
Threshold	T1	T2				
Value	5.1	7.8				

2.7 Revise the position and velocity for each particle

The position X_{ind}^{ite} and velocity V_{ind}^{ite} for a particle in the ite^{th} iteration are revised based on the Eqs. (1) and (2). In addition, the range [0, 10] for the position X_{ind}^{ite} and the range [−1.5, 1.5] for the velocity V_{ind}^{ite} are used to check their feasibility. If the position or velocity violates the range constraints, the maximal (or minimal) value is used to substitute for the violated position or velocity.

$$V_{ind}^{ite} = w \times V_{ind}^{ite-1} + c_1 \times rand1 \times \left(Pbest_{ind} - X_{ind}^{ite-1}\right) + c_2 \times rand2 \times \left(Gbest - X_{ind}^{ite-1}\right) \quad (1)$$

$$X_{ind}^{ite} = X_{ind}^{ite-1} + V_{ind}^{ite} \quad (2)$$

2.8 Parameter setting for the proposed hybrid heuristic method

There are six parameters which are adopted and need to be considered for this proposed hybrid heuristic method: *np* (particle number), I (iteration number), w (inertia weight), c_1, c_2 (learning factor), and *nlns* (the iteration number for LNS). According to Shi and Eberhart [22] and Jordehi and Jasni [10], linear decreasing inertia weight is adopted ($w = \frac{(0.9-0.4)*(I-ite)}{I} + 0.4$), c_1 is set 2, and c_2 is set 2. The other values are experimentally determined based on the minimal average cost. *np* is tried from 30 to 50 (30, 40, 50). I is tried from100 to 300 (100, 200, 300). *nlns* is tried from 3 to 7 (3, 5, 7).

3 Experimental results

3.1 Results

In order to examine the computational effectiveness and efficiency of the proposed heuristic method (H_1), three methods are compared to the proposed method. The first method used is the heuristic method (H_2) proposed by Qin et al. [21]. The second method is a heuristic method (H_3) proposed by Zhang et al. [29]. The third method is PSO (H_4) (the H_4 is almost the same as H_1. However, the *Apply large neighborhood search to Gbest* procedure is deleted from H_1). The heuristic methods are coded using DEV C++, and the tests are carried out on a PC2.6 GHz under WIN7, Intel Core Q8400, and 4-GB RAM.

A set of ten test instances is adopted based on the original ten data sets given by Qin et al. [21]. All the instances involve a planning horizon of seven time units, and the retailer set size varies from 30 to 210, with an interval of 20 units geographically

Table 5 Best solutions and CPU times for different heuristic methods

Instance	H_1		H_2		H_3		H_4	
	Best cost	CPU time (s)	Best cost	CPU time (s)	Best cost	CPU time (s)	Best cost	CPU time (s)
p_30_7	33,284.2	184	34,764.3	196	33,898.3	88	33,632.3	176
p_50_7	51,142.9	325	54,255.8	271	53,453.9	238	51,513.9	310
p_70_7	54,852.7	542	60,822.6	466	56,663.1	288	55,627.4	508
p_90_7	71,008.9	797	77,659.9	689	73,281.3	786	72,534.1	721
p_110_7	78,715.2	847	87,951.4	738	79,786.7	1025	79,253.8	793
p_130_7	88,694.2	1237	98,777.8	1051	89,527.5	1588	89,436.6	1192
p_150_7	119,237.6	1741	136,673	1517	123,493.9	1897	120,852.2	1627
p_170_7	149,872.3	2237	168,469	1921	157,274.9	1966	153,092.5	2148
p_190_7	179,692.6	2605	205,384	2375	184,316.3	2246	182,737.2	2582
p_210_7	202,901.7	2782	229,354	2221	206,507.4	2748	205,419.8	2721
Average	102,940.2	1330	115,411.2	1144.5	105,820.3	1287	104,410.0	1278

dispersed around the supplier. The distance matrix is obtained by calculating the Euclidean distance. The customer demand rate is modified to vary in each time unit, while the holding cost remains the same, and a new vehicle fleet and capacity are adjusted for the new demand. As for the parameter, *np*, *I*, and *nlns*, for the proposed method H_1, the value is determined based on the minimal cost criterion after experiments as follows: $np=50$, $I=200$, and $nlns=5$.

Table 5 shows the best solutions in ten runs and their corresponding CPU times for different heuristic methods H_1, H_2, H_3, and H_4. The table shows that H1 is better than H_2, H_3, and H_4 in terms of cost in all instances since H_1 takes advantage of PSO and LNS to effectively find a better solution (the average cost for H_1 is 102,940.2 smaller than 115,411.2 for H_2, 105, 820.3 for H_3, and 104,410.0 for H_4). In addition, when the problem size increases, the CPU time increases since the search space increases for H_1, H_2, H_3, and H_4. Although H_1 takes a little more time than other heuristic methods since a local search is applied to each particle in different iterations and a LNS is applied to the global best solution *Gbest*, it is still acceptable (the average CPU time for H_1 is 1330 s larger than 1144.5 s for H_2, 1287 s for H_3, and 1278 s for H_4). Table 6 shows the average costs and gap ($=(H_i-\min\{H_1, H_2, H_4\})/\min\{H_1, H_2, H_4\}*100\ \%$) for H_1, H_2, and H_4 (the average costs for H_3 are not available in [29]). This table shows that H_1 is better than H_2 and H_4 since H_1 takes advantage of PSO and LNS to effectively find a better solution (the average cost [gap] for H_1 is 103,179.7 and is smaller than 115,917.0 [10.93 %] for H_3 and 105,097.8 [1.86 %] for H_4).

In addition to the methods used for the PIRP, the threshold value determined in the proposed method and period size are two important factors that influence the solution quality. Table 7

Table 6 Average costs for different heuristic methods

Instance	H_1		H_2		H_4	
	Average cost	GAP[a] (%)	Average cost	GAP (%)	Average cost	GAP (%)
p_30_7	33,712.9	0	34,868.2	3.43	33,724.2	0.03
p_50_7	51,238.7	0	54,437.5	6.24	51,647.6	0.80
p_70_7	55,063.8	0	61,013.7	10.81	55,804.3	1.34
p_90_7	71,255.3	0	78,035.1	9.51	72,691.8	2.02
p_110_7	78,832.4	0	88,269.3	11.97	80,340.1	1.91
p_130_7	88,724.7	0	99,216.2	11.82	90,208.2	1.67
p_150_7	119,763.9	0	137,305.6	14.65	122,387.9	2.19
p_170_7	150,106.7	0	169,277.3	12.77	153,401.7	2.20
p_190_7	180,016.4	0	206,487.9	14.71	183,560.5	1.97
p_210_7	203,082.5	0	230,259.6	13.38	207,211.4	2.03
Average	103,179.7	0	115,917.0	10.93	105,097.8	1.86

The average costs for H_3 are not available in Zhang et al. [29]

[a] GAP (%)=$(H_i-\min\{H_1, H_2, H_4\})/\min\{H_1, H_2, H_4\}*100\ \%$

Table 7 Average costs for different threshold value setting in H_1

Instance	Threshold value determined by PSO	Threshold value (=0)
p_30_7	33,712.9	33,720.4
p_50_7	51,238.7	51,486.5
p_70_7	55,063.8	55,418.1
p_90_7	71,255.3	71,641.8
p_110_7	78,832.4	79,274.3
p_130_7	88,724.7	89,322.4
p_150_7	119,763.9	120,581.7
p_170_7	150,106.7	151,095.2
p_190_7	180,016.4	181,139.6
p_210_7	203,082.5	204,511.4

shows the results in H_1 with different threshold values. The H_1 adopting the threshold value determined by PSO is better than that equal to 0 in terms of average cost. Since different threshold values perform variedly in different problems (from smaller size to larger size), PSO can find better values than a fixed threshold value of 0. Table 8 shows that the period size decreases from 7 to 5 (selected from the first five of seven periods adopted by [21]), and the average cost decreases. The average cost (transportation and inventory cost) decreases since the period size increases and customer demand increases.

3.2 Practical and academic implications

Since PIRP is an NP-hard problem, different search approaches have been proposed for solving the PIRP [21, 29]. Although most of the studies have provided effective heuristic methods, such as TS and SA, it is necessary to continually explore the global optimal solution to help manufacturers or organizations decrease transportation and inventory costs and increase their benefits. Recently, PSO has become a promising method; however, it is easily trapped in local optimality. This paper proposed a hybrid heuristic method integrating PSO and LNS to improve the performance of pure heuristic methods and to overcome the drawbacks of pure heuristic methods. Additionally, the proposed hybrid heuristic method could explore better solutions than those existing hybrid heuristic methods in terms of average cost. Therefore, manufacturers or future studies should consider the above conceptions and the hybrid heuristic method to improve their competitive advantage.

Table 8 Average costs for different period sizes in H_1

Instance	Period size=7	Period size=5
p_30	33,712.9	25,872.1
p_50	51,238.7	42,039.4
p_70	55,063.8	46,812.6
p_90	71,255.3	60,182.3
p_110	78,832.4	66,219.7
p_130	88,724.7	74,819.8
p_150	119,763.9	102,872.9
p_170	150,106.7	131,538.2
p_190	180,016.4	156,091.5
p_210	203,082.5	181,012.4

4 Conclusions

The purpose of this paper was to adopt a hybrid heuristic method to search for the optimal solution for the PIRP. The results are in line with previous studies: hybrid heuristic methods generate better results than single heuristic methods (PSO). The hybrid heuristic method, combining PSO and LNS, for the PIRP is 10.93 % better than the existing method (H_2) and 1.86 % better than the pure heuristic method (H_4) in terms of average cost. Furthermore, two important factors, threshold value and period size, were analyzed for sensitivity analysis. The experimental results also indicated that the method proposed by this paper had better results for dealing with the PIRP.

As for future research directions, there are two issues that require further attention: (1) More factors should be considered, such as time window constraint, pricing, location allocation, etc.; (2) better search algorithms should be developed to help academic or practical industries find solutions easily and fast. We are hopeful that future research will offer more detailed results and efficient search algorithms. We also recommend that the hybrid heuristic method in this paper be applied to other IRPs.

Acknowledgments This work was partially supported by the Ministry of Science and Technology, Taiwan, R OC, under grant NSC 100-2221-E-020-019 and MOST 104-2221-E-020-013. The authors thank the anonymous reviewers for their helpful comments and suggestions.

Appendix (adopted from Qin et al. [21])

Notations

T: The length of planning horizon
N: The number of retailers
K: The number of vehicles
t: The index for time units ($1 \le t \le T$)
p: The index for time units ($1 \le p \le$ T)
i: The index for retailers ($1 \le i \le N$); 0 means the supplier
j: The index for retailers ($1 \le j \le N$)
k: The index for vehicle ($1 \le k \le K$)
Q: The vehicle capacity
d_{it}: The demand of retailer i on time unit t

h_i: The holding cost at retailer i per unit of product
c_{ij}: The transportation cost directly from retailer i to retailer j
dp_{ipt}: The quantity of the demand d_{it} that is satisfied by the delivery on time unit p
y_{ipt}: If dp_{ipt}>0, y_{ipt}=1; otherwise, y_{ipt}=0
x_{ijtk}: If retailer i immediately precedes retailer j on vehicle k at time t, x_{ijtk}=1; otherwise, x_{ijtk}=0
q_{it}: The quantity that delivered to retailer i at time t

The PIRP model is as follows:

$$\min C_t = \sum_{i\in N, i\neq 0} \sum_{t\in T} \sum_{P=0}^{T} \left[\frac{dq_{ipt} h_i |t-p| + dq_{ipt} h_i (\sum_{z<p} dq_{ipt})}{d_{it} + \frac{1}{2} dq_{ipt} h_i \left(\frac{dq_{ipt}}{d_{it}} \right)} \right]$$

$$+ \sum_{t\in T} \sum_{k\in K} \sum_{i\in N} \sum_{j\in N} C_{ij} x_{ijtk}$$

Subject to

$$\sum_{i\neq j, i\in N} x_{ijtk} - \sum_{p\neq j, p\in N} x_{jptk} = 0, \forall j\in N, \forall k\in K, \forall t\in T \quad \text{(A1)}$$

$$\sum_{k\in K} \sum_{j\in N} x_{ijtk} \leq 1, \forall i\in N/\{0\}, \forall t\in T \quad \text{(A2)}$$

$$\sum_{i\in B} \sum_{j\in B} x_{ijtk} \leq |B| - 1, \forall k\in K, \forall t\in T, B\subseteq N/\{0\}, |B| > 1 \quad \text{(A3)}$$

$$\sum_{i\in N, j\neq i} \sum_{j\in N0} x_{ijtk} q_{jt} \leq Q, \forall k\in K, \forall t\in T \quad \text{(A4)}$$

$$\sum_{t\in T} q_{it} = \sum_{t\in T} d_{it}, \forall i\in N/\{0\} \quad \text{(A5)}$$

$$q_{it} \leq \mathrm{M} \sum_{j>i, j\in N} \sum_{k\in K} x_{ijtk}, \forall i\in N, \forall t\in T \quad \text{(A6)}$$

$$\sum_{j\in N, j\neq i} \sum_{k\in K} x_{ijtk} \leq M q_{it}, \forall i\in N, \forall t\in T \quad \text{(A7)}$$

$$\sum_{p} dq_{ipt} = d_{it}, \forall i\in N, \forall t\in T \quad \text{(A8)}$$

$$\sum_{t} dq_{ipt} = q_{it}, \forall i\in N, \forall p\in T \quad \text{(A9)}$$

$$y_{ipt} \leq M dq_{ipt}, \forall i\in N, \forall p, \forall t\in T \quad \text{(A10)}$$

$$M y_{ipt} \geq dq_{ipt}, \forall i\in N, \forall p, \forall t\in T \quad \text{(A11)}$$

$$\sum_{P\in B} \sum_{t\in B} y_{ipt} \leq |B| - 1, \forall i\in N, B\subseteq T, |B| > 1 \quad \text{(A12)}$$

$$0 \leq dq_{ipt};\ q_{it} \geq 0;\ d_{it} \geq 0;\ x_{ijtk} \in [0, 1], \forall i\in N, \forall j\in N, \forall t, P\in T, \forall k\in K \quad \text{(A13)}$$

In the above formulation, the objective function is to minimize the sum of transportation cost and inventory cost. Constraint (A1) insures that every point entered by the vehicle should be the same point that the vehicle leaves. Constraint (A2) insures that if a vehicle arrives at retailer j on time t by vehicle k, then the vehicle k must depart retailer j on the same day, while each retailer is served by at most one vehicle in a time unit. Constraint (A3) is a sub-tour elimination constraint. Constraint (A4) states that the amount of each route to retailers must be less than or equal to vehicle capacity. Constraint (A5) guarantees that the delivery quantity satisfies the demand exactly of each retailer over the horizon. The following two constraints (A6) and (A7) give the relationship between delivery routes and the delivered quantity of retailer i. They show that if no vehicle serves retailer i during this service time, there should be no product delivered to i, and once a quantity is delivered to retailer i, there is one and only one vehicle that serves the retailer. Constraint (A8) guarantees that the sum of the amount of products that is responsible for demand on time t equals its demand for each retailer. Constraint (A9) ensures that the sum of the amount of products that is responsible for each time of each retailer on time t equals its delivered quantity. Constraints (A10) and (A11) give the relationship between response relationship and response quantity. Constraint (A12) is a sub-tour elimination constraint of the response relationship. Constraint (A13) is the constraint for decision variables.

References

1. Anderson H, Hoff A, Christiansen M, Hasle G, Lokketangen A (2010) Industrial aspect and literature survey: combined inventory management and routing. Comput Oper Res 37:1515–1536
2. Chih M (2015) Self-adaptive check and repair operator-based particle swarm optimization for the multidimensional knapsack problem. Appl Soft Comput 26:378–389
3. Choong F, Phon-Amnuaisuk S, Alias MY (2011) Metaheuristic methods in hybrid flow shop scheduling problem. Expert Syst Appl 38:10787–10793
4. Federgruen A, Zipkin P (1984) A combined vehicle routing and inventory allocation problem. Oper Res 32:1019–1037
5. Fu C, Fu Z (2010) Optimization on stochastic inventory-routing problem in multi-cycle two-echelon system. Comput Eng Appl 46:198–201
6. Gaur V, Fisher ML (2004) A periodic inventory routing problem at a supermarket chain. Oper Res 52:813–822
7. Golden B, Assad A, Dahl R (1984) Analysis of a large scale vehicle routing problem with an inventory component. Large Scale Syst 7: 181–190
8. Hansen P, Mladenović N (2001) Variable neighborhood search: principles and applications. Eur J Oper Res 130:449–467
9. Hansen P, Mladenović N (2010) Variable neighborhood search: methods and applications. Ann Oper Res 175:367–407
10. Jordehi AR, Jasni J (2013) Parameter selection in particle swarm optimisation: a survey. J Exp Theoret Artif Intell 24:527–542
11. Kiranyaz S, Pulkkinen J, Gabbouj M (2011) Multi-dimensional particle swarm optimization in dynamic environments. Expert Syst Appl 38:2212–2223
12. Küçükoğlu İ, Öztürk N (2015) An advanced hybrid meta-heuristic algorithm for the vehicle routing problem with backhauls and time windows. Comput Ind Eng 86:60–68
13. Lee YH, Jung JW, Lee KM (2006) Vehicle routing scheduling for cross-docking in the supply chain. Comput Ind Eng 51:247–256
14. Li JQ, Pan QK, Liang YC (2010) An effective hybrid tabu search algorithm for multi-objective flexible job-shop scheduling problems. Comput Ind Eng 59:647–662

15. Liu S-C, Chen A-Z (2012) Variable neighborhood search for the inventory routing and scheduling problem in a supply chain. Expert Syst Appl 39:4149–4159
16. Liu S-C, Lin C-C (2005) A heuristic method for the combined location routing and inventory problem. Int J Adv Manuf Technol 26:372–381
17. Marinakis Y, Marinaki M, Dounias G (2010) A hybrid particle swarm optimization algorithm for the vehicle routing problem. Eng Appl Artif Intell 23:463–472
18. Mirabi M (2014) A hybrid electromagnetism algorithm for multi-depot periodic vehicle routing problem. Int J Adv Manuf Technol 71:509–518
19. Miyamoto Y, Kubo M (2001) Case study: the inventory routing for vending machines. J Oper Res Soc Jpn 44:378–389
20. Moin NH, Salhi S (2007) Inventory routing problems: a logistical overview. J Oper Res Soc 58:1185–1194
21. Qin L, Miao L, Qing FR, Zhang Y (2014) A local search method for periodic inventory routing problem. Expert Syst Appl 41:765–778
22. Shi Y, Eberhart R (1998) A modified particle swarm optimizer. IEEE World Congress on Computational Intelligence, Anchorage, AK, pp 69–73
23. Sindhuchao S, Romeijn HE, Akcali E, Boondiskulchok R (2005) An integrated inventory-routing system for multi-item joint replenishment with limited vehicle capacity. J Glob Optim 32:93–118
24. Tasgetiren MF, Liang Y-C, Sevkli M, Gencyilmaz G (2007) A particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem. Eur J Oper Res 177:1930–1947
25. Vidal T, Crainic TG, Gendreau M, Prins C (2013) Heuristics for multi-attribute vehicle routing problems: a survey and synthesis. Eur J Oper Res 231:1–21
26. Wang Y, Ma X, Xu M, Liu Y, Wang Y (2015) Two-echelon logistics distribution region partitioning problem based on a hybrid particle swarm optimization–genetic algorithm. Expert Syst Appl 42:5019–5031
27. Wei L, Zhang Z, Zhang D, Lim A (2015) A variable neighborhood search for the capacitated vehicle routing problem with two-dimensional loading constraints. Eur J Oper Res 243:798–814
28. Zachariadis EE, Tarantilis CD, Kiranoudis CT (2014) An integrated local search method for inventory and routing decisions. Expert Syst Appl 36:10239–10248
29. Zhang Y, Qi M, Miao L, Liu E (2014) Hybrid metaheuristic solutions to inventory location routing problem. Transp Res E 70:305–323
30. Zhao Q, Wang SY, Lai KK (2007) A partition approach to the inventory/routing problem. Eur J Oper Res 177:786–802
31. Zhao Q, Chen S, Zang C (2008) Model and algorithm for inventory/routing decision in a three-echelon logistics system. Eur J Oper Res 191:623–635