OR Spectrum (2005) 27: 569–581
DOI: 10.1007/s00291-005-0196-8

OR Spectrum

The joint replenishment problem with quantity discounts under constant demand*

B.C. Cha and I.K. Moon

Department of Industrial Engineering, Pusan National University, Busan 609-735, Korea
(e-mail: ikmoon@pusan.ac.kr)

Abstract. In many practical situations quantity discounts on basic purchase price exist, and taking advantage of these can result in substantial savings. Quantity discounts have been considered in many production and inventory models. But unlike other research areas, there have been no studies to quantity discounts in the joint replenishment problem. The purpose of this paper is to develop efficient algorithms for solving this problem. Firstly, we suggest useful propositions to develop efficient heuristic algorithms. Secondly, we develop two algorithms using these propositions. Numerical examples are shown to illustrate the procedures of these algorithms. Extensive computational experiments are performed to analyze the effectiveness of the heuristics.

Keywords: Joint replenishment problem – Quantity discounts

1 Introduction

For the inventory system with multiple items, cost savings can be obtained when the replenishment of several items are coordinated. The joint replenishment problem (JRP) is the multi-item inventory problem of coordinating the replenishment of a group of items that may be jointly ordered from a single supplier. In this situation, the ordering cost has two components – a major common ordering cost S incurred whenever an order is placed and a minor ordering cost s_i incurred if item i is incurred in the order. In the deterministic joint replenishment problem, it is assumed that the

* The authors are grateful to the editor and two anonymous referees for their valuable comments on earlier versions of this paper. This work was supported by "Research Center for Logistics Information Technology (LIT)" hosted by the Ministry of Education and Human Resources Development in Korea.
Correspondence to: I.K. Moon

major ordering cost is charged at a basic cycle time T and that the ordering cycle of each item is some integer multiple of this basic cycle.

Over the last few decades, the joint replenishment problem has received much attention. Arkin et al. [1] proved that the JRP is an NP-hard problem, i.e., the JRP is not solvable by polynomial-time algorithms. Goyal and Satir [10] reviewed various modeling and experimentation under deterministic and stochastic demand conditions. Goyal [7] proposed an enumeration approach, and he claimed that it always secures a global optimal solution. However, Van Eijs [24] has pointed out that the lower bound on an optimal cycle time used by Goyal [7] does not guarantee a global optimal solution and derived another algorithm that improves Goyal's algorithm [7]. Unlike these enumeration approaches, Silver [20, 21] discussed the advantages and disadvantages of coordinating replenishments and presented a simple non-iterative procedure to solve it. Kaspi and Rosenblatt [16] proposed an approach based on trying several values of the basic cycle time between a minimum and a maximum value. Then they applied the heuristic of Kaspi and Rosenblatt [15] for each value of the basic cycle time, which is a modified version of the algorithm of Silver [20]. They showed that their procedure (RAND) outperforms all the available heuristics. Later, Goyal and Deshmukh [9] proposed an improved lower bound used by Kaspi and Rosenblatt [16]. Wildeman et al. [25] presented a new optimal solution approach based on Lipschitz optimization to obtain a solution with an arbitrarily small deviation from the optimal value. Khouja et al. [17] developed a genetic algorithm and compared it with RAND. In spite of its inferiority to RAND, they discussed the advantages of the genetic algorithm in terms of the ability of handling constrained problems. Li [18] considered the multi-buyer joint replenishment problem and proposed a new efficient RAND method.

The general JRP models assume that the unit cost is constant, no matter what quantity is purchased. But in reality, suppliers may induce their customers to place larger orders by offering them quantity discounts. If the quantity purchased is greater than a specified "price break" quantity, the cost per unit is reduced. Two types of price break schedule can be considered (all-units and incremental discount schedule). The all-units discount applies the discounted price to all units beginning with the first unit, if the quantity purchased exceeds the price break quantity. The incremental discount schedule applies the discounted price only to those units over the price break quantity. It is common practice to include this discount policy in the published price schedule.

These quantity discounts have been considered in many production and inventory models (Silver et al. [22]). Pirkul and Aras [19] considered the problem of multiple item EOQs with all-units discounts offered separately for each item. Güder et al. [13] studied the incremental quantity discounts case for the problem of Pirkul and Aras [19]. The study of Güder et al. [13] is based on the independent cycle approach. Güder and Zydiak [11, 12] presented a non-stationary method and fixed cycle approach modifying above independent cycle approach. Hariri et al. [14] presented a geometric programming approach instead of the traditional Lagrangian method to handle a resource constraint for the all-units quantity discounts schedule. Benton [2] presented an efficient heuristic algorithm for evaluating alternative discount schedules under conditions of multi-item, multi-supplier considering the resource limitations. However, the coordination of the replenishments is not considered in these studies. Benton and Park [3] classified the literature on lot sizing determination under several types of discount schemes and discussed some of the significant literature in this area.

For the joint replenishment problem, Chakravarty [4] proposed the grouping procedure. He considered the group discounts available on the total purchase value of a group replenishment. However, as mentioned in Silver et al. [22], sometimes discounts are offered not for the total volume of a replenishment made up of several different items but for each individual item included in the replenishment. Chung et al. [5] presented a mathematical programming for both all-units and incremental quantity discounts schedules and developed an effective heuristic algorithm for the incremental quantity discounts schedule under a dynamic demand condition. Xu et al. [26] developed an algorithm based on a dynamic programming for the all-units quantity discounts schedule under a dynamic demand condition. Van der Duyn Schouten et al. [23] proposed a heuristic method to incorporate a quantity discounts schedule in the framework of can-order strategies under a stochastic demand condition.

However, there has been no research on dealing with the quantity discounts of each item for the joint replenishment problem under constant demand. The purpose of this paper is to develop an efficient algorithm for solving the JRP considering the quantity discounts of each item. We consider the all-units discount schedule. The following section introduces this problem and suggests propositions to be utilized in developing solution procedures. The simple heuristic algorithm for solving this problem has been developed in Section 3 and we illustrate the procedure of the algorithm using a numerical example. In Section 4, we show how the modified RAND algorithm can be used to handle the JRP with quantity discounts. Computational experiments are performed to analyze the effectiveness of the heuristics in Section 5. Finally, we summarize the present work.

2 The joint replenishment problem with quantity discounts

Similar to the general joint replenishment problem under a deterministic demand condition, the following assumptions are made:

1. The demand rate for each item is constant and deterministic.
2. The replenishment lead time is of known duration.
3. Shortages are not allowed.
4. The entire order quantity is delivered at the same time.
5. The price of each item is dependent on the magnitude of the replenishment of each item. All-units discount schedule is considered in this paper.
6. The inventory holding cost for each item is known and constant, independent of the price of each item.

Moreover, to discuss the joint replenishment problem with quantity discounts, we introduce the following notation:

i: index of item, $i = 1, 2, \cdots., n$

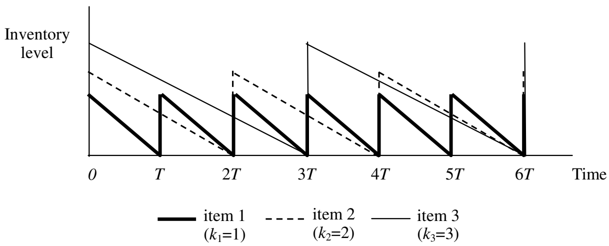

Fig. 1. Behavior of inventory over time

y: index of price break
D_i: demand rate of item i
S: major ordering cost
s_i: minor ordering cost of item i
h_i: inventory holding cost of item i, per unit per unit time
p_{iy}: price of item i in the y th price break
q_{iy}: quantity of item i triggering the y th price break
T: basic cycle time (decision variable)
k_i: integer number that decides the replenishment quantity of item i (decision variable)

A joint replenishment is made every T time intervals. However, all items may not be included in each replenishment. Item i is only included every k_iT time intervals. This means that the replenishment of each item is made at every integer multiple (k_i) of the group replenishment time interval (T) as shown in Figure 1. This also indicates that k_iT is the cycle time of item i.

According to the above assumptions and definitions, the total relevant cost per unit time to be minimized is as follows:

$$TC(T, k_1, k_2, \cdots, k_n) = \frac{S + \sum_{i=1}^{n} \frac{s_i}{k_i}}{t} + \sum_{i=1}^{n} \frac{D_i k_i T h_i}{2} + \sum_{i=1}^{n} C_i(t, k_i) D_i$$

where C_i is the unit cost function of item i. This is a step function of T and k_i. For the all-units quantity discount, the unit cost C_i is represented as follows:

$$C_i(T, k_i) = p_{iy}, \quad \text{for} \;\; q_{iy} \leq D_i k_i T < q_{i(y+1)}$$

where D_ik_iT is the order quantity Q_i of item i.

To find T and k_i s that minimize the total relevant cost per unit time, we propose the following two propositions.

Proposition 1. For a given set of k_i's, the optimal basic cycle time T is $T^* = \text{argmin}_{T_y}\{TC(T_y)\}$, where T_y includes T^0 and all T_{iy} satisfying the following condition:

$$T_{iy} = \frac{q_{iy}}{D_i k_i} > T^0 = \sqrt{2\left(S + \sum_{i=1}^{n} \frac{s_i}{k_i}\right) \Big/ \sum_{i=1}^{n} D_i k_i h_i}$$

T^0 means the optimal T obtained from the first order derivative of the total cost function of the JRP ignoring quantity discounts.

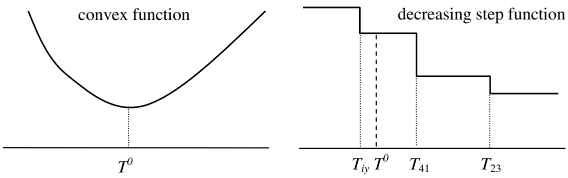

Fig. 2. Graphs of $TC_1(T)$ and $\sum_{i=1}^{n} C_i(T)D_i$

Proof. For a given set of k_i's, the total cost function of this problem is as follows.

$$TC(T) = \frac{S + \sum_{i=1}^{n} \frac{s_i}{k_i}}{T} + \sum_{i=1}^{n} \frac{D_i k_i T h_i}{2} + \sum_{i=1}^{n} C_i(T) D_i$$
$$= TC_1(T) + \sum_{i=1}^{n} C_i(T) D_i$$

where it is obvious that $TC_1(T)$ is a convex function and $\sum_{i=1}^{n} C_i(T)D_i$ is a decreasing step function. If T^0 is the value that minimizes $TC_1(T), TC(T^0)$ is always less than $TC(T)$ for $T < T^0$ because $TC_1(T) > TC_1(T^0)$ and $\sum_{i=1}^{n} C_i(T)D_i > \sum_{i=1}^{n} C_i(T^0)D_i$. For $T > T^0$, $TC_1(T)$ also increases as T increases. However, $TC(T^0)$ is not always less than $TC(T)$ because $\sum_{i=1}^{n} C_i(T)D_i$ decreases at the price break points of T. Therefore, the optimal basic cycle time T minimizing TC is T^0 or one of the price break points T_{iy} that is larger than T^0 (See Fig. 2).

Proposition 2. For a given T, the optimal integer value of k_i is $k_i^* = \text{argmin}_{k_{iy}}\{TC(k_{iy})\}$, where k_{iy} includes k_i^0 and all k_{iy} satisfying the following condition:

$$k_{iy} = \left\lceil \frac{q_{iy}}{D_i T} \right\rceil > k_i^0$$

where k_i^0 means the optimal value of k_i for the JRP ignoring quantity discounts. It is well known that the value can be obtained from the following optimal conditions (Goyal [6]).

$$k_i^0 \left(k_i^0 - 1\right) \le \frac{2s_i}{D_i h_i T^2} \le k_i^0 \left(k_i^0 + 1\right)$$

Proof. For a given T, the total cost function of this problem is as follows.

$$TC(k_1, k_2, \cdots, k_n) = \frac{S + \sum_{i=1}^{n} \frac{s_i}{k_i}}{T} + \sum_{i=1}^{n} \frac{D_i k_i T h_i}{2} + \sum_{i=1}^{n} C_i(k_i) D_i$$
$$= \frac{S}{T} + \sum_{i=1}^{n} \left[TC_2(k_i) + C_i(k_i) D_i\right]$$

where $TC_2(k_i) = \frac{s_i}{Tk_i} + \frac{D_i k_i T h_i}{2}$.

If we assume that the value of k_i is a real number, we know that $TC_2(k_i)$ is a convex function in k_i and $C_i(k_i)D_i$ is a decreasing step function. Moreover, k_i is independent of k_j $(j \neq i)$. Though k_i is an integer, it is obvious that

$TC(k_i^0)$ is always less than $TC(k_i)$ for $k_i < k_i^0$ because $TC_2(k_i) > TC_2(k_i^0)$ and $C_i(k_i)D_i > C_i(k_i^0)D_i$. For $k_i > k_i^0$, $TC_2(k_i)$ also increases as k_i increases. However, $TC(k_i^0)$ is not always less than $TC(k_i)$ because $C_i(k_i)D_i$ decreases at the price break points of k_i. Therefore, the optimal value of k_i minimizing TC is k_i^0 or one of the price break points (k_{iy}) that is larger than k_i^0.

Remark 1. By Proposition 2, it is obvious that $TC(k_i^0) - TC(k_{iy}) > 0$ if the optimal value of k_i exists at the price break point satisfying $k_{iy} > k_i^0$. Therefore, we can easily find the optimal value of k_i by checking the following values at all the price break points satisfying $k_{iy} > k_i^0$.

$$TC\left(k_i^0\right) - TC\left(k_{iy}\right) = \frac{s_i}{T}\left(\frac{1}{k_i^0} - \frac{1}{k_{iy}}\right) + \frac{D_i T h_i}{2}\left(k_i^0 - k_{iy}\right) + D_i\left[C_i(k_i^0) - C_i(k_{iy})\right]$$

3 The simple heuristic algorithm (SH)

Using the two propositions and Remark 1 in the previous section, we now develop a simple recursive algorithm to solve the JRP with quantity discounts. The procedure of this heuristic algorithm is as follows.

The simple heuristic algorithm (SH)

(Step 1) Set the iteration number $r = 0$.
Put $T(r) = 0$ and $(k_1(r), k_2(r), \cdots, k_n(r)) = 1$ and go to Step 2.

(Step 2) Set $r = r + 1$.
For a given set of $k_i(r)$s, find the optimal value of T using Proposition 1.
Set $T(r) = T$. If $T(r) = T(r-1)$, stop. Otherwise, go to Step 3.

(Step 3) For a given value of $T(r)$, find optimal values of k_i for each item i using Proposition 2 and Remark 1.
Set $k_i(r) = k_i$ for each item i and go to Step 2.

To illustrate the SH, we solve a numerical example. We transform the example problem of Goyal [8] to consider quantity discounts. The data for this example are given in Table 1. We also assume $S = \$200$ and all $h_i = \$1$.

Solution of the example

In Step 1, $r = 0, T(0) = 0, (k_1(0), k_2(0), \cdots, k_6(0)) = (1, 1, 1, 1, 1, 1)$ and we go to Step 2. In Step 2, for a given set of $(k_1(0), k_2(0), \cdots, k_6(0)) = (1, 1, 1, 1, 1, 1)$, $T^0(1) = 0.2188$ and $TC(T^0) = \$125{,}932$. TCs for all price break points that are larger than $T^0(1)$ are as follows.

By Proposition 1, $T(1) = 0.2188$ and we go to Step 3. In Step 3, for a given $T(1) = 0.2188$, $(k_1^0(1), k_2^0(1), \cdots, k_6^0(1)) = (1, 1, 1, 1, 2, 3)$. TCs for all price break points that are larger than $(k_1^0(1), k_2^0(1), \cdots, k_6^0(1))$ are shown in Table 3.

Table 1. Data for the example

Item i	D_i	s_i	Price break	Price
1	10,000	45	$Q_1 < 500$	6.25
			$500 \leq Q_1 < 1,000$	6.20
			$1,000 \leq Q_1 < 2,000$	6.15
			$Q_1 \geq 2,000$	6.10
2	5,000	46	$Q_2 < 500$	6.25
			$500 \leq Q_2 < 1,000$	6.20
			$1,000 \leq Q_2 < 2,000$	6.15
			$Q_2 \geq 2,000$	6.10
3	3,000	47	$Q_3 < 500$	6.25
			$500 \leq Q_3 < 1,000$	6.20
			$Q_3 \geq 1,000$	6.15
4	1,000	44	$Q_4 < 500$	6.25
			$Q_4 \geq 500$	6.20
5	600	45	$Q_5 < 300$	6.25
			$Q_5 \geq 300$	6.20
6	200	47	$Q_6 < 150$	6.25
			$Q_6 \geq 150$	6.20

Table 2. Calculations using Proposition 1

Item i	D_i	$k_i(0)$	q_{iy}	Price	$q_{iy}/D_ik_i(0)$	TC
2	5,000	1	2,000	6.10	0.4000	\$126,345
3	3,000	1	1,000	6.15	0.3333	\$126,172
4	1,000	1	500	6.20	0.5000	\$127,018
5	600	1	300	6.20	0.5000	\$127,018
6	200	1	150	6.20	0.7500	\$129,167

By Proposition 2 and Remark 1, $(k_1(1), k_2(1), \cdots, k_6(1)) = (1, 1, 1, 1, 2, 4)$. Thus, we go to Step 2. In Step 2, for a given set of $(k_1(1), k_2(1), \cdots, k_6(1)) = (1, 1, 1, 1, 2, 4), T^0(2) = 0.1991$ and $TC(T^0) =$ \$126,521. TCs for all price break points that are larger than $T^0(2)$ are shown in Table 4.

By Proposition 1, $T(2) = 0.2000$ and we go to Step 3 as $T(2) \neq T(1)$. In Step 3, for a given $T(2) = 0.2000$, $(k_1^0(2), k_2^0(2), \cdots, k_6^0(2)) = (1, 1, 1, 2, 2, 3)$. TCs for all price break points that are larger than $(k_1^0(2), k_2^0(2), \cdots, k_6^0(2))$ are shown in Table 5.

By Proposition 2 and Remark 1, $(k_1(2), k_2(2), \cdots, k_6(2)) = (1, 1, 1, 2, 3, 4)$, and we go to Step 2. In Step 2, for a given set of $(k_1(2), k_2(2), \cdots, k_6(2)) =$

Table 3. Calculations using Proposition 2 and Remark 1

Item i	D_i	q_{iy}	Price	$k_{iy} = \lceil q_{iy}/D_iT(1)\rceil$	$TC(k_i^0) - TC\,(k_{iy})$
2	5,000	2,000	6.10	2	−191.88
3	3,000	1,000	6.15	2	−70.80
4	1,000	500	6.20	3	−34.74
5	600	300	6.20	3	−1.36
6	200	150	6.20	**4**	**6.02**

Table 4. Calculations using Proposition 1

Item i	D_i	$k_i(1)$	q_{iy}	Price	$q_{iy}/D_ik_i(1)$	TC
1	10,000	1	2,000	6.10	0.2000	$125,771
2	5,000	1	1,000	6.15	0.2000	$125,771
			2,000	6.10	0.4000	$126,401
3	3,000	1	1,000	6.15	0.3333	$126,159
4	1,000	1	500	6.20	0.5000	$127,193
5	600	2	300	6.20	0.2500	$125,850

Table 5. Calculations using Proposition 2 and Remark 1

Item i	D_i	q_{iy}	Price	$k_{iy} = \lceil q_{iy}/D_iT(2)\rceil$	$TC(k_i^0) - TC\,(k_{iy})$
2	5,000	2,000	6.10	2	−135.00
3	3,000	1,000	6.15	2	−32.50
4	1,000	500	6.20	3	−13.33
5	600	300	6.20	**3**	**7.50**
6	200	150	6.20	**4**	**9.58**

$(1,1,1,2,3,4), T^0(3) = 0.1850$ and $TC(T^0) =$ $126,501. TCs for all price break points that are larger than $T^0(3)$ are shown in Table 6.

By Proposition 1, $T(3) = 0.2000$ and we stop iterations since $T(3) = T(2)$. Table 7 shows the optimal policies of the general JRP and the JRP with quantity discounts for this example.

Table 6. Calculations using Proposition 1

Item i	D_i	$k_i(2)$	q_{iy}	Price	$q_{iy}/D_i k_i(2)$	TC
1	10,000	1	2,000	6.10	0.2000	\$125,754
2	5,000	1	1,000	6.15	0.2000	\$125,754
			2,000	6.10	0.4000	\$126,597
3	3,000	1	1,000	6.15	0.3333	\$126,287
4	1,000	2	500	6.20	0.2500	\$125,882
6	200	4	150	6.20	0.1875	\$126,491

Table 7. Comparison between the general JRP and the JRP with quantity discounts

	T	k_is
general JRP	0.1893	1,1,1,2,2,3
JRP with quantity discounts	0.2000	1,1,1,2,3,4

4 The quantity discount RAND algorithm

In the JRPs, the iterative algorithms such as the SH generally converge to local optimal solutions and a global optimum is not guaranteed. To overcome this problem, Kaspi and Rosenblatt [16] developed the RAND algorithm. They found several local optimal solutions from the iterative algorithm using many different first values of T and obtained the best solution among all the local optimal solutions. They have shown that the first value of T greatly influences to find the optimal solution by the simulation study. Using this idea, we modify RAND and develop a new algorithm for solving the JRP with quantity discounts. The modified RAND algorithm, we call QD-RAND from now on, is as follows.

The quantity discount RAND algorithm (QD-RAND)

(Step 1) Compute

$$T_{\max} = \max \left[\sqrt{2\left(S + \sum_{i=1}^{n} s_i\right) \Big/ \sum_{i=1}^{n} D_i h_i}, \frac{\max\{q_{iy}\}}{D_i} \right] \quad \text{and}$$

$$T_{\min} = \min \left(\sqrt{\frac{2s_i}{D_i h_i}} \right)$$

for all values of item i, where $\max\{q_{iy}\}$ means the largest price break point of item i.

(Step 2) Divide the range $[T_{\min}, T_{\max}]$ into m different equally spaced values of $T(T_1, \cdots, T_j, \cdots, T_m)$. The value of m is to be decided by the decision maker.

Table 8. Iterations by QD-RAND

T_j	Iteration r	k_i in Step 4	T in Step 5	TC_j
$T_1 = 0.0949$	1	3,3,2,3,6,8	0.0976	
	2	3,3,2,3,6,8	0.0976	$127,337
$T_2 = 0.2587$	1	1,1,1,2,2,3	0.2000	
	2	1,1,1,2,3,4	0.2000	$125,754
$T_3 = 0.4224$	1	1,1,1,1,1,2	0.2122	
	2	1,1,1,1,3,4	0.2000	
	3	1,1,1,2,3,4	0.2000	$125,754
$T_4 = 0.5862$	1	1,1,1,1,1,1	0.2188	
	2	1,1,1,1,2,4	0.2000	
	3	1,1,1,2,3,4	0.2000	$125,754
$T_5 = 0.7500$	1	1,1,1,1,1,1	0.2188	
	2	1,1,1,1,2,4	0.2000	
	3	1,1,1,2,3,4	0.2000	$125,754

Table 9. The discount schedules

Price break	Price
$Q_i < 500$	p_{i1}
$500 \leq Q_i < 1,000$	$0.99p_{i1}$
$1,000 \leq Q_i < 2,000$	$0.98p_{i1}$
$Q_i \geq 2,000$	$0.97p_{i1}$

Set $j = 0$.

(Step 3) Set $j = j + 1$ and $r = 0$.

(Step 4) Set $r = r + 1$.
For a given value of $T_j(r)$, find optimal values of $k_i(r)$ for each item i using Proposition 2 and Remark 1.

(Step 5) For a given set of $k_i(r)$s, find the optimal value of T using Proposition 1.

(Step 6) Set $T_j(r) = T$. If $T_j(r) \neq T_j(r-1)$, go to Step 4.
Otherwise, Set $T_j^* = T_j(r)$, $k_{ij}^* = k_i^*(r)$ for each item i and compute TC_j for this $(T_j^*, k_{ij}^* s)$.
If $j = m$, stop and select $(T_j^*, k_{ij}^* s)$ with the minimum TC.
Otherwise, go to Step 3.

To illustrate QD-RAND, we use the previous numerical example. We use m=5.

It turns out that QD-RAND gives the same solution as the simple heuristic algorithm. Table 8 also shows that the algorithm converges to the different local optimum for different initial values of T. For T_4 and T_5, we can see that the iterations of QD-RAND are completely the same as that of SH.

Table 10. Computational results (% improvement)

			Number of minimum				% improvement									
			QD-RAND						QD-RAND							
n	S	SH	$m = 0.5n$	$m = n$	$m = 2n$	$m = 4n$	SH Max	SH Avg.	$m = 0.5n$ Max	$m = 0.5n$ Avg.	$m = n$ Max	$m = n$ Avg.	$m = 2n$ Max	$m = 2n$ Avg.	$m = 4n$ Max	$m = 4n$ Avg.
10	25	6	57	58	68	100	0.297	0.111	0.119	0.018	0.119	0.016	0.119	0.011	–	–
	50	16	33	36	50	100	0.229	0.068	0.142	0.034	0.142	0.032	0.142	0.021	–	–
	75	37	40	43	56	100	0.213	0.041	0.213	0.035	0.213	0.031	0.119	0.018	–	–
	100	54	55	59	71	100	0.198	0.025	0.198	0.024	0.198	0.020	0.112	0.011	–	–
20	25	0	71	71	81	100	0.304	0.171	0.087	0.010	0.087	0.010	0.073	0.006	–	–
	50	1	45	45	62	100	0.264	0.136	0.123	0.022	0.123	0.020	0.092	0.010	–	–
	75	0	18	22	45	100	0.226	0.112	0.137	0.038	0.115	0.034	0.104	0.018	–	–
	100	3	12	16	47	100	0.208	0.090	0.166	0.052	0.127	0.043	0.127	0.022	–	–
30	25	0	80	80	84	100	0.325	0.209	0.047	0.003	0.047	0.003	0.044	0.003	–	–
	50	0	61	61	67	100	0.273	0.179	0.060	0.008	0.060	0.008	0.043	0.006	–	–
	75	0	43	44	54	100	0.245	0.153	0.073	0.015	0.073	0.014	0.061	0.010	–	–
	100	0	31	33	49	100	0.226	0.130	0.086	0.026	0.086	0.023	0.085	0.013	–	–
50	25	0	88	88	89	100	0.330	0.246	0.030	0.002	0.030	0.002	0.030	0.001	–	–
	50	0	80	80	84	100	0.307	0.225	0.038	0.003	0.038	0.003	0.037	0.002	–	–
	75	0	65	65	77	100	0.284	0.265	0.047	0.005	0.047	0.005	0.045	0.003	–	–
	100	0	51	51	67	100	0.265	0.186	0.056	0.009	0.052	0.009	0.052	0.004	–	–
Max.							0.330		0.213		0.213		0.142			
Avg.		7	52	53	66	100		0.147		0.019		0.017		0.010		

5 Computational experiments

In this section we compare the performances of the two algorithms for a number of randomly generated JRPs with quantity discounts. The initial price (p_{i1}), demand rate, minor ordering cost and holding cost are generated from uniform distribution on the ranges [5, 25] , [500, 5000], [10, 50] and $[0.1p_{i1}, 0.2p_{i1}]$, respectively. The discount schedules are defined in Table 9.

Four different values of n (10, 20, 30 and 50) and four different values of S (25, 50, 75 and 100) are considered. For each combination of n and S, 100 problems are generated and solved using both SH and QD-RAND for a total of 1600 problems. To compare the effectiveness of the QR-RAND for different values of m, four different values of ($m = 0.5n, n, 2n$, and $4n$) are used. A summary of computational results is shown in Table 10.

As shown in Table 10, we can confirm that the performance of QR-RAND is improved as the value of m is increased. This result is consistent with the experiments of Kaspi and Rosenblatt [16]. For the larger values of n ($n =$ 30 and 50), the value of m is sufficient to consider only $0.5n(m = 15$ and $25)$ as the maximum error found is only 0.086%. For the smaller values of n ($n = 10$ and 20), $4n(m = 40$ and $80)$ or larger values of m can be considered since the computational time is very small.

6 Concluding remarks

We presented two efficient algorithms for solving the joint replenishment problem with quantity discounts. From extensive computational experiments, we show that the performance of QR-RAND is improved as the value of m is increased. These algorithms can be easily adopted by purchasing managers who are responsible for the coordination of the replenishment. There are three possible extensions which might be interesting research problems. Firstly, an incremental discount scheme can be applied to this problem. Secondly, the model can be extended to multiple suppliers case in which different suppliers offer different discount prices. Thirdly, a model in which the inventory holding cost depends on the product value can be developed.

References

1. Arkin E, Joneja D, Roundy R (1989) Computational complexity of uncapacitated multi-echelon production planning problems. Operations Research Letters 8: 61–66
2. Benton W (1991) Quantity discounts under conditions of multiple items, multiple suppliers and resource limitations. International Journal of Production Research 29: 1953–1961
3. Benton W, Park S (1996) A classification of literature on determining the lot size under quantity discounts. European Journal of Operational Research 92: 219–238
4. Chakravarty A (1984) Joint inventory replenishments with group discounts based on invoice value. Management Science 30: 1105–1112

5. Chung C, Hum S, Kirca O (1996) The coordinated replenishment dynamic lot-sizing problem with quantity discounts. European Journal of Operational Research 94: 122–133
6. Goyal S (1973) Determination of economic packaging frequency for items jointly replenished. Management Science 20: 232–238
7. Goyal S (1974) Determination of optimum packaging frequency of items jointly replenished. Management Science 23: 436–443
8. Goyal S (1975) Analysis of joint replenishment inventory systems with resource restriction. Operations Research Quarterly 26: 197–203
9. Goyal S, Deshmukh S (1993) A note on the economic ordering quantity for jointly replenished items. International Journal of Production Research 31: 2959–2961
10. Goyal S, Satir A (1989) Joint replenishment inventory control: deterministic and stochastic models. European Journal of Operational Research 38: 2–13
11. Güder F, Zydiak J (1997) Non-stationary ordering policies for multi-item inventory systems subject to a single resource constraint and quantity discounts. Computers & Operations Research 24: 61–71
12. Güder F, Zydiak J (2000) Fixed cycle ordering policies for capacitated multiple item inventory systems with quantity discounts. Computers & Industrial Engineering 38: 67–77
13. Güder F, Zydiak J, Chaudhry S (1994) Capacitated multiple item ordering with incremental quantity discounts. Journal of the Operational Research Society 45: 1197–1205
14. Hariri A, Abou-el-ata M, Goyal S (1995) The resource constrained multi-item inventory problem with price discount: a geometric programming approach. Production Planning & Control 6: 374–377
15. Kaspi M, Rosenblatt M (1983) An improvement of Silver's algorithm for the joint replenishment problem. IIE Transactions 15: 264–269
16. Kaspi M, Rosenblatt M (1991) On the economic ordering quantity for jointly replenished items. International Journal of Production Research 29: 107–114
17. Khouja M, Michalewicz Z, Satoskar S (2000) A comparison between genetic algorithms and the RAND method for solving the joint replenishment problem. Production Planning & Control 11: 556–564
18. Li Q (2004) Solving the multi-buyer joint replenishment problem with the RAND method. Computers & Industrial Engineering 46: 755–762
19. Pirkul H, Aras O (1985) Capacitated multiple item ordering problem with quantity discounts. IIE Transactions 17:206–211
20. Silver E (1975) Modifying the economic order quantity (EOQ) to handle coordinated replenishment of two or more items. Production & Inventory Management 16: 26–38
21. Silver E (1976) A simple method of determining order quantities in jointly replenishments under deterministic demand. Management Science 22: 1351–1361
22. Silver E, Pyke D, Peterson R (1998) Inventory management and production planning and scheduling, 3rd edn. Wiley, New York
23. Van der Duyn Schouten F, Van Eijs M, Heuts R (1994) Coordinated replenishment systems with discount opportunities. International Journal of Production Research 32: 2879–2895
24. Van Eijs M (1993) A note on the joint replenishment problem under constant demand. Journal of the Operational Research Society 44: 185–191
25. Wildeman R, Frenk J, Dekker R (1997) An efficient optimal solution method for the joint replenishment problem. European Journal of Operational Research 99: 433–444
26. Xu J, Lu L, Glover F (2000) The deterministic multi-item dynamic lot size problem with joint business volume discount. Annals of Operations Research 96: 317–337