

This article was downloaded by: [Professor Seyed Taghi Akhavan Niaki]

On: 08 September 2014, At: 10:57

Publisher: Taylor & Francis

Informa Ltd Registered in England and Wales Registered Number: 1072954 Registered office: Mortimer House, 37-41 Mortimer Street, London W1T 3JH, UK



Production & Manufacturing Research: An Open Access Journal

Publication details, including instructions for authors and subscription information:

<http://www.tandfonline.com/loi/tpmr20>

A multiproduct EOQ model with inflation, discount, and permissible delay in payments under shortage and limited warehouse space

Seyed Hamid Reza Pasandideh^a, Seyed Taghi Akhavan Niaki^b & Behzad Maleki Vishkaei^c

^a Faculty of Engineering, Department of Industrial Engineering, Kharazmi University, Tehran, Iran

^b Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran

^c Young Researchers and Elite Club, Qazvin Branch, Islamic Azad University, Qazvin, Iran

Published online: 04 Sep 2014.

To cite this article: Seyed Hamid Reza Pasandideh, Seyed Taghi Akhavan Niaki & Behzad Maleki Vishkaei (2014) A multiproduct EOQ model with inflation, discount, and permissible delay in payments under shortage and limited warehouse space, *Production & Manufacturing Research: An Open Access Journal*, 2:1, 641-657

To link to this article: <http://dx.doi.org/10.1080/21693277.2014.955214>

PLEASE SCROLL DOWN FOR ARTICLE

Taylor & Francis makes every effort to ensure the accuracy of all the information (the "Content") contained in the publications on our platform. Taylor & Francis, our agents, and our licensors make no representations or warranties whatsoever as to the accuracy, completeness, or suitability for any purpose of the Content. Versions of published Taylor & Francis and Routledge Open articles and Taylor & Francis and Routledge Open Select articles posted to institutional or subject repositories or any other third-party website are without warranty from Taylor & Francis of any kind, either expressed or implied, including, but not limited to, warranties of merchantability, fitness for a particular purpose, or non-infringement. Any opinions and views expressed in this article are the opinions and views of the authors, and are not the views of or endorsed by Taylor & Francis. The accuracy of the Content should not be relied upon and should be independently verified with primary sources of information. Taylor & Francis shall not be

liable for any losses, actions, claims, proceedings, demands, costs, expenses, damages, and other liabilities whatsoever or howsoever caused arising directly or indirectly in connection with, in relation to or arising out of the use of the Content.

This article may be used for research, teaching, and private study purposes. Terms & Conditions of access and use can be found at <http://www.tandfonline.com/page/terms-and-conditions>

It is essential that you check the license status of any given Open and Open Select article to confirm conditions of access and use.

A multiproduct EOQ model with inflation, discount, and permissible delay in payments under shortage and limited warehouse space

Seyed Hamid Reza Pasandideh^a, Seyed Taghi Akhavan Niaki^{b*} and Behzad Maleki Vishkaei^c

^aFaculty of Engineering, Department of Industrial Engineering, Kharazmi University, Tehran, Iran; ^bDepartment of Industrial Engineering, Sharif University of Technology, Tehran, Iran; ^cYoung Researchers and Elite Club, Qazvin Branch, Islamic Azad University, Qazvin, Iran

(Received 21 October 2013; accepted 12 August 2014)

This study considers a multiproduct economic order quantity problem where delay in payment is permissible and the retailer can benefit cash discounts. The amount of discount and the length of the grace period depend on the order quantity and all the costs increase by an inflation rate. Moreover, the shortage is backlogged and the limited warehouse space leads to a constraint for storage. We first formulate the problem into a non-linear integer-programming model and then we propose a hybrid genetic algorithm and simulated annealing (GA+SA) to solve it. Since there is no benchmark available in the literature, a GA is developed as well to validate the results obtained. The parameters of both algorithms are tuned using the Taguchi method. Finally, numerical examples are solved to evaluate the performances and to compare the efficiency of the two solution procedures.

Keywords: multiproduct EOQ; inflation; discount; permissible delay in payments; shortage; limited warehouse space; genetic algorithm; simulated annealing

1. Introduction and literature review

The economic order quantity (EOQ) and the economic production quantity are the most applicable models in inventory control environments, where many researchers have studied them under different conditions using various assumptions. Two of the assumptions that are important in attracting new customers while maintaining the old ones are the permissible delay in payments and the cash discounts. Most of the times the suppliers use these politics to encourage the retailers to buy more merchandises. However, the retailers always cannot buy as much as they want because of their limited resources such as budget and warehouse capacity.

In many businesses, the credit has a great role in financing. As the customer does not pay any interest during the permissible delay in payment time, the permissible delay policy can be viewed as a type of price discount. Moreover, it provides an economic chance for the retailer to delay the payment up to the last day of the permissible delay period. However, suppliers usually consider a delay cost for the retailers who pay after the permissible delay time to prevent payments after the grace period.

Goyal (1985) developed mathematical models to obtain EOQ of an item for which the supplier permits a fixed delay in setting the amount owned to him. Roy and

*Corresponding author. Email: niaki@sharif.edu

Samanta (2011) extended Goyal's (1985) model to include unequal unit selling and purchasing prices. Abad and Jaggi (2003) considered an inventory model under credit period in which the end demand was price-sensitive. Moreover, both the credit period and the price were considered seller's decision variables. They assumed short-time capital cost and gain rates for the buyer, where there was no cash discount for settling the account early. Huang (2007) studied an EOQ model under permissible delay in payment. The main difference of his research with the previously published research works was to consider a partial delay in payments when the order quantity is less than the amount of quantity that leads to fully delayed payments. In this case, as the order is filled in, the retailer must make a partial payment to the supplier. The remaining purchasing cost must be paid off at the end of the trade credit period. In the case of a full delay in payment, all the payment occurs at the end of the credit period. Sana and Chaudhuri (2008) considered an inventory model under permissible delay time and discount to maximize the profit, where the amount of discount depended on the length of the grace time. Ouyang, Teng, Goyal, and Yang (2009) presented an EOQ model under deterioration rate and partial permissible delay time. In their research, when the order quantity is less than a predetermined quantity for a fully delayed payment, the retailer must pay the partial payment by taking a loan with an interest charged per dollar per year. The constant sale revenue would pay off the loan.

The all-units and the incremental discount policies are the two general types of discount that are used for cost reductions. On the one hand, if the discount is given on all units of a product that is purchased in quantity more than a predetermined breakpoint, it is called the all-units discount policy. On the other hand, if the discount is applied only for the units that are purchased beyond a given breakpoint at which the discount is applied, it is called the incremental discount policy. Shinn, Hwang, and Park (1996) studied an EOQ model in which the ordering cost included a fixed ordering and a freight costs. By assuming the freight cost having a quantity discount, they solved the problem under permissible delay in payments. Matsuyama (2001) considered an EOQ model in which both the purchasing and the setup cost depended on the order quantity. While the discount price was assumed a decreasing function of the ordering quantity, an increasing function was used for the setup cost. The goal was to find the ordering cycle and the ordering quantity to maximize the one-day's average profit. Bhaba and AL-Kind (2006) considered an EOQ model with a sale period in which there were two kinds of regular and special order quantities during the period. They compared different discount scenarios to sense the effect of different parameters on the ordering policies. Mendoza and Ventura (2008) presented exact algorithms to solve an EOQ model that includes truckload and less than truckload transportation costs under both all-units and incremental discounts. Toptal (2009) used all-units discount for replenishment decisions considering stepwise freight cost. Munson and Hu (2010) used both all-units and incremental discount policies to find the optimal ordering quantities that would minimize the total cost. The developed model of their problem was solved under different centralization scenarios.

Leopoldo, Neale, and Goyal (2010) developed an EOQ model in which the shortage was backlogged with two types of backordering costs. Further, three different scenarios for one-time discount were considered. In the first scenario, the discount was offered at a reorder point that applies to all units. In the second one, the discount was offered at a reorder point and applied only to the units to be purchased. Finally, in the third scenario, the discount was offered between reorder points and applied only to the additional units.

An inflation rate can be assumed in EOQ to have a more realistic model. Hariga (1995) developed a model that includes both inflationary trends and time discounting, and compared it with the standard EOQ model. Liao, Tsai, and Su (2000) studied the effects of the inflation and deterioration rates in an EOQ model with permissible delay in payment where an initial-stock-dependent consumption rate was considered. They solved the model based on the relation between the credit period and the cycle time. More relevant inventory works can be seen in Bather (1966), Chao (1992), Brander, Levén, and Segerstedt (2005), and Borgonovo (2010).

In this paper, a multiproduct EOQ problem with a permissible delay in payment and discount (all-units and incremental), both depending on the order quantity, is considered. While the shortage is backlogged and the warehouse has a limited capacity, there is a penalty cost per time on the payments that occur after the permissible delay time.

The organization of the rest of the paper is as follows. In the next section, the problem is defined. The mathematical formulation of the problem is given in Section 3. In Section 4, both a genetic algorithm (GA) and a hybrid GA and simulated annealing (SA) algorithm (GA+SA) are proposed to solve the model. In Section 5, the Taguchi approach is used to tune the parameters of the algorithms. In order to demonstrate the applications of the proposed modeling and to validate and compare their performances, some numerical examples are solved in Section 6. Finally, the conclusion and recommendations for future researches come in Section 7.

2. Problem definition

Consider a company (retailer) that works with a supplier. The company stores several products replenished by the supplier to satisfy its customers' needs. Based on the contract between the supplier and the company, the retailer is provided permissible delay in payments and discount, both depending on the order quantity. If the payment is not made during the grace period, not only the company has to pay all the purchase cost without any discount, but also it has to pay an additional penalty cost for the lateness. The penalty depends on the length of the lateness. This agreement encourages the company to pay during the grace period with a benefit of flexible payments. The other specifications of the problem are defined as follows.

- The constant demand rates of the products are known.
- The lead time is neglected.
- Replenishments are instantaneous.
- The inflation rate is constant.
- When an order is received, defective items are rejected after a 100% inspection process.
- The company (retailer) pays the purchasing cost to the supplier when the entire product in his warehouse is sold out and places a new order at this time. In other words, for a certain received product the payment cannot occur before its inventory level reaches zero. Although this assumption restricts the application of the proposed modeling, it is made to simplify the derivations.
- Shortages are allowed and unsatisfied demands are fully backlogged.
- All the costs increase by an inflation rate that is fixed during the replenishment period. In other words, the costs are determined at the beginning of each period based on its corresponding periodical inflation rate.

- Some of the products benefit from the all-units discount policy and the others from the incremental discount policy.
- The capacity of the warehouse is limited.

A graph of the inventory position of the i th product is illustrated in Figure 1, where the notations are given in Section 3.1. The objective is to determine the order quantity and the shortage of each product such that the total inventory cost is minimized while the warehouse constraint is satisfied.

3. Problem modeling

To formulate the problem, the parameters are first defined in Section 3.1. Different inventory costs are then derived in Section 3.2. Finally, the mathematical model of the problem is developed in Section 3.3.

3.1. Notation

For $i = 1, 2, \dots, n$, the parameters and the decision variables of the model are defined as follows.

Parameters

- H Length of planning horizon (in this paper, it is considered one year; $H = 1$)
 n Number of products
 D_i Demand rate of product i (number of units per period)
 r Constant inflation rate
 l Holding cost rate
 $A_i(t)$ Ordering cost at time t for product i . That is $A_i(t) = A_i e^{rt}$, where A_i is the ordering cost at time zero
 $\pi_i(t)$ Backordering cost per unit at time t for product i . That is $\pi_i(t) = \pi_i e^{rt}$, where π_i is the backordering cost per unit at time zero

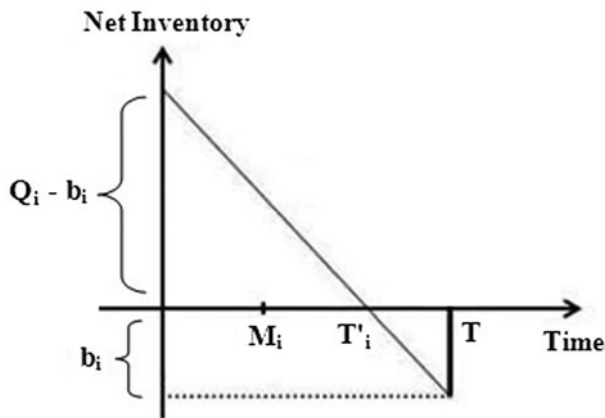


Figure 1. A graph of the inventory position of product i .

- $\varpi_i(t)$ Backordering cost per unit per unit of time at time t for product i . That is $\varpi_i(t) = \varpi_i e^{rt}$, where ϖ_i is the backordering cost per unit per unit of time at time zero
- $\gamma_i(t)$ Delay cost per unit of time at time t for product i . That is $\gamma_i(t) = \gamma_i e^{rt}$, where γ_i is the delay cost per unit of time at time zero
- T_i Cycle time of product i
- N_i Number of cycle times of product i ($N_i = \frac{H}{T_i}$)
- T'_i Part of the cycle time in which there is an inventory for product i
- K Number of products that benefit all-units discount
- M_i Permissible delay period for paying to the supplier the purchasing cost of product i (a function of Q_i)
- $C_i(t)$ Purchasing price per unit of product i at time t (a function of Q_i). That is $C_i(t) = C_i e^{rt}$, where C_i is the purchasing cost per unit at time zero
- $C_{i,j}$ Purchasing price per unit of product i at the j th discount point $j=1, 2, \dots, m+1$
- $M_{i,j}$ Permissible delay time for product i at the j th discount point
- f_i Space that is needed to store one unit of product i
- F Total available warehouse space
- F' The warehouse space that is needed to store all the products' order quantities
- p_i Average fraction of an order quantity of product i , that is not defective
- TS_i Total annual ordering cost of product i
- TB_i Total annual shortage cost product i
- TH_i Total annual holding cost of product i
- TM_i Total annual delay cost of product i
- TP Total TP'_i discounted purchasing cost of product i
- TP''_i Total purchasing cost of product i without discount
- TP_i Total annual purchasing cost of product i
- TC Total annual cost of all products

Decision variables

- Q_i Order quantity of product i
- b_i Maximum shortage (backorder) level of product i
- $x_{i,j}$ Binary variables used to calculate $C_{i,j}$ in Table 1
- $y_{i,j}$ Binary variables used to obtain $M_{i,j}$ in Table 1
- $h_i(t)$ Holding cost per unit per unit of time at time t for product i . That is $h_i(t) = h_i e^{rt}$, where h_i is the holding cost per unit per unit of time at time zero and depends on variables Q_i and s_i defined as

Table 1. Grace period and purchasing price as a function of the order size ($q_{i,0} = 0$).

Grace period calculation		Price calculation	
Q_i	M_i	$p_i Q_i$	C_i
$0 < Q_i \leq q_{i,1}$	$M_{i,1}$	$0 < p_i Q_i \leq q_{i,1}$	$C_{i,1}$
$q_{i,1} < Q_i \leq q_{i,2}$	$M_{i,2}$	$q_{i,1} < p_i Q_i \leq q_{i,2}$	$C_{i,2}$
...
$q_{i,m-1} < Q_i \leq q_{i,m}$	$M_{i,m}$	$q_{i,m-1} < p_i Q_i \leq q_{i,m}$	$C_{i,m}$
$q_{i,m} < Q_i < u$	$M_{i,m+1}$	$q_{i,m} < p_i Q_i \leq u$	$C_{i,m+1}$

Downloaded by [Professor Seyed Taghi Akhavan Niaki] at 10:57 08 September 2014

$$s_i = \begin{cases} 1; & \text{If product } i \text{ receives discount } (TM_i = 0) \\ 0; & \text{Otherwise } (TM_i > 0) \end{cases}$$

Assuming the supplier offers a longer permissible delay time for larger order sizes and gains a larger discount to purchase more products, Table 1 shows the relationships among the ordering quantity, the permissible delay time, and the unit-selling cost, where u represents a big number. In this table, if $q_{i,0} = 0 < Q_i \leq q_{i,1}$ for an example, then the permissible delay time equals to $M_{i,1}$ with a unit selling price of $C_{i,1}$.

3.2. Costs derivation

The annual system cost involves ordering, holding, shortage, delay, and purchasing costs derived as follows.

3.2.1. Annual ordering cost (TS_i)

The total annual ordering cost of product i depends on the inflation rate and the number of ordering cycles and is obtained by:

$$TS_i = A_i(0) + A_i(T_i) + \dots + A_i((N_i - 1)T_i) = \sum_{t=0}^{N_i-1} A_i e^{rtT_i} = A_i \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \quad (1)$$

3.2.2. Annual holding cost (TH_i)

As the holding cost per unit per unit of time ($h_i(t)$) depends on the inflation rate, the number of cycles per year, and the area under the inventory graph of Figure 1, the annual holding cost can be derived using:

$$TH_i = \sum_{t=0}^{N_i-1} T'_i \frac{(p_i Q_i - b_i)}{2} h_i(tT_i) = \sum_{t=0}^{N_i-1} T'_i \frac{(p_i Q_i - b_i)}{2} h_i e^{rtT_i} \quad (2)$$

Now, based on Figure 1 we have,

$$T'_i = \frac{P_i Q_i - b_i}{D_i} \quad (3)$$

Inserting Equation (3) in (2) leads to:

$$TH_i = \frac{(p_i Q_i - b_i)^2}{2D_i} h_i \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \quad (4)$$

3.2.3. Annual shortage cost (TB_i)

Shortages are backlogged, where their cost is considered time-dependent and time-independent components. On the one hand, the time-independent annual shortage hinges on the shortage in each cycle, the number of cycles, and the inflation rate. On the other hand, the time-dependent shortage cost depends entirely on the area of the shortage graph, the number of cycles per year, and the inflation rate. As a result, the first term of Equation (5) denotes the time-independent and the second one refers to the time-dependent shortage cost.

$$TB_i = \sum_{t=0}^{N_i-1} b_i \pi_i e^{rT_i} + \sum_{t=0}^{N_i-1} (T_i - T'_i) \frac{b_i}{2} \varpi_i e^{rT_i} = \left(b_i \pi_i + (T_i - T'_i) \frac{b_i}{2} \varpi_i \right) \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \tag{5}$$

However, referring to Figure 1, we have:

$$T_i - T'_i = \frac{b_i}{D_i} \tag{6}$$

Then, replacing Equation (6) in (5) results in:

$$TB_i = \left(b_i \pi_i + \frac{b_i^2}{2D_i} \varpi_i \right) \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \tag{7}$$

3.2.4. Annual delay cost (TM_i)

Since it is assumed the company is not able to pay to the supplier before selling all units of the product, if the inventory level reaches zero after the grace period, there will be a penalty cost. Hence, the total annual delay cost is derived under two different circumstances in the following two subsections.

3.2.4.1. $M_i \geq T'_i$. In this case, the inventory level reaches zero before the end of the grace period. Thus, the total delay cost equals to zero. In other words:

$$TM_i = 0 \tag{8}$$

3.2.4.2. $M_i < T'_i$. In this case, the inventory level reaches zero after the grace period. As a result, the delay cost hinges on the length of the interval between M_i and T'_i , number of cycles per year, and the inflation rate. In other words, we have

$$TM_i = \sum_{t=0}^{N_i-1} (T'_i - M_i) \gamma_i (tT_i) = (T'_i - M_i) \gamma_i \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \tag{9}$$

Replacing Equation (3) in (9) results in:

$$TM_i = \left(\frac{P_i Q_i - b_i}{D_i} - M_i \right) \gamma_i \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \tag{10}$$

Then, based on what was derived in subsection 3.2.3.1 and 3.2.3.2, the annual total delay cost is obtained as:

$$TM_i = \max \left\{ 0, \left(\frac{p_i Q_i - b_i}{D_i} - M_i \right) \gamma_i \right\} \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \tag{11}$$

where M_i is found in Table 1.

Using the binary variables $y_{i,j}$, M_i can be replaced by $\sum_{j=1}^{m+1} y_{i,j} M_{i,j}$ considering $\sum_{j=1}^{m+1} y_{i,j} = 1$. Hence, the total annual delay cost is:

$$TM_i = \max \left\{ 0, \left(\frac{p_i Q_i - b_i}{D_i} - \sum_{j=1}^{m+1} y_{i,j} M_{i,j} \right) \gamma_i \right\} \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) = Z_i \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \tag{12}$$

3.2.5. Annual purchasing cost (TP_i)

Based on what we assumed in section two, some of the products receive all-units discount and the others benefit from incremental discount policy. Further, if the payment is not made during the grace period, the company has to pay all the purchasing cost without any discount. Without loss of generality, to derive the total annual purchasing cost, assume the first K products receive all-units and the rest get incremental discount.

Since the unit purchase cost of $C_{i,j}$ is found in Table 1, using binary variables $x_{i,j}$, C_i can be replaced by $\sum_{j=1}^{m+1} x_{i,j}C_{i,j}$ under the condition $\sum_{j=1}^{m+1} x_{i,j} = 1$. Furthermore, since the annual total purchasing cost depends on the order quantity, the number of cycles per year, the inflation rate, and the purchase cost per unit, and knowing that when $M_i \geq T'_i$ the company benefits discount, the purchasing cost can be obtained as follows.

For the first K products that receive all-units discount, i.e. $i = 1, 2, \dots, K$, we have:

$$TP'_i = \sum_{t=1}^{N_i-1} p_i Q_i C_i (tT_i) = P_i Q_i C_i \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) = \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \sum_{j=1}^{m+1} P_i Q_i x_{i,j} C_{i,j} \quad (13)$$

For the rest of the $n - K$ items, i.e. $i = K + 1, \dots, n$, the incremental discount is employed. Thus,

$$\begin{aligned} TP'_i &= \sum_{t=0}^{N_i-1} \left(\sum_{j=2}^{m+1} \left((p_i Q_i - q_{i,j-1}) C_{i,j} e^{rtT_i} + \sum_{f=1}^{j-1} (q_{i,j-f} - q_{i,j-f-1}) C_{i,j-f} e^{rtT_i} \right) x_{i,j} + \right. \\ &\quad \left. p_i Q_i C_{i,1} e^{rtT_i} x_{i,1} \right) \\ &= \sum_{t=0}^{N_i-1} e^{rtT_i} \left(\sum_{j=2}^{m+1} \left((p_i Q_i - q_{i,j-1}) C_{i,j} + \sum_{f=1}^{j-1} (q_{i,j-f} - q_{i,j-f-1}) C_{i,j-f} \right) x_{i,j} + \right. \\ &\quad \left. p_i Q_i C_{i,1} x_{i,1} \right) \\ &= \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \left(\sum_{j=2}^{m+1} \left((p_i Q_i - q_{i,j-1}) C_{i,j} + \sum_{f=1}^{j-1} (q_{i,j-f} - q_{i,j-f-1}) C_{i,j-f} \right) x_{i,j} \right. \\ &\quad \left. + p_i Q_i C_{i,1} x_{i,1} \right) \end{aligned} \quad (14)$$

However, in the case where $M_i < T'_i$, the company does not receive discount and the annual total purchasing cost is obtained as follows.

$$TP''_i = \sum_{t=0}^{N_i-1} p_i Q_i C_{i,1} e^{rtT_i} = p_i Q_i C_{i,1} \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right); \quad i = 1, 2, \dots, n \quad (15)$$

Finally, the annual total purchasing cost can be obtained using the following equation.

$$TP_i = TP'_i s_i + TP''_i (1 - s_i); \quad i = 1, 2, \dots, n \quad (16)$$

3.3. Annual total cost (TC)

Based on Equations (1), (4), (7), (12), and (16) the annual total cost of all products is obtained using Equation (17).

$$\begin{aligned}
 TC &= \sum_{i=1}^n (TS_i + TH_i + TB_i + TM_i + TP_i) \\
 &= \sum_{i=1}^n \left(\left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \left(A_i + \frac{(p_i Q_i - b_i)^2}{2D_i} h_i + b_i \pi_i + \frac{b_i^2}{2D_i} \varpi_i + Z_i \right) \right. \\
 &\quad \left. + TP'_i s_i + TP''_i (1 - s_i) \right) \tag{17}
 \end{aligned}$$

3.4. The mathematical model

Based on what we derived in the previous subsections and knowing the objective is to minimize the total system cost, the mathematical model of the problem at hand becomes:

$$\begin{aligned}
 TC &= \sum_{i=1}^n \left(\left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \left(A_i + \frac{(p_i Q_i - b_i)^2}{2D_i} h_i + b_i \pi_i + \frac{b_i^2}{2D_i} \varpi_i + Z_i \right) \right. \\
 &\quad \left. + TP''_i s_i + TP'_i (1 - s_i) \right) \tag{18}
 \end{aligned}$$

s.t.:

$$TP'_i = \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \sum_{j=1}^{m+1} P_i Q_i x_{i,j} C_{i,j} ; \quad i = 1, 2, \dots, k \tag{19}$$

$$\begin{aligned}
 TP'_i &= \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) \left(\sum_{j=2}^{m+1} \left((p_i Q_i - q_{i,j-1}) C_{i,j} + \sum_{f=1}^{j-1} (q_{i,j-f} - q_{i,j-f-1}) C_{i,j-f} \right) x_{i,j} \right. \\
 &\quad \left. + p_i Q_i C_{i,1} x_{i,1} \right) ; \text{ for } i = k + 1, \dots, n \tag{20}
 \end{aligned}$$

$$TP''_i = \sum_{t=0}^{N_i-1} p_i Q_i C_{i,1} e^{rtT_i} = p_i Q_i C_{i,1} \left(\frac{e^{rH} - 1}{e^{rT_i} - 1} \right) ; \quad i = 1, 2, \dots, n \tag{21}$$

$$h_i = \left(l \times \frac{TP'_i}{p_i Q_i} \times \left(\frac{e^{rT_i} - 1}{e^{rH} - 1} \right) \right) s_i + (l \times C_{i,1} \times (1 - s_i)) ; \quad i = 1, 2, \dots, n \tag{22}$$

$$\sum_{j=0}^m q_{i,j} y_{i,j+1} \leq Q_i \leq \sum_{j=1}^m q_{i,j} y_{i,j} + u y_{i,m+1} ; \quad i = 1, 2, \dots, n \tag{23}$$

$$\sum_{j=0}^m q_{i,j} x_{i,j+1} \leq p_i Q_i \leq \sum_{j=1}^m q_{i,j} x_{i,j} + u x_{i,m+1} ; \quad i = 1, 2, \dots, n \tag{24}$$

$$\sum_{j=1}^{m+1} y_{i,j} = 1 ; \quad i = 1, 2, \dots, n \tag{25}$$

$$\sum_{j=1}^{m+1} x_{ij} = 1; \quad i = 1, 2, \dots, n \quad (26)$$

$$\sum_{i=1}^n f_i(p_i Q_i - b_i) \leq F \quad (27)$$

$$Z_i \geq \left(\frac{p_i Q_i - b_i}{D_i} - \sum_{j=1}^{m+1} y_{ij} M_{ij} \right) \gamma_i; \quad i = 1, 2, \dots, n \quad (28)$$

$$Z_i \times s_i = 0; \quad i = 1, 2, \dots, n \quad (29)$$

$$Z_i + s_i > 0; \quad i = 1, 2, \dots, n \quad (30)$$

$$x_{ij} = 0, 1, \quad y_{ij} = 0, 1, \quad s_i = 0, 1, \quad Z_i \geq 0, \quad Q_i \geq 0, \quad b_i \geq 0, \quad i = 1, 2, \dots, n; \quad (31)$$

$$j = 1, 2, \dots, m + 1$$

Some brief notes on the constraints of the model follows.

- As the holding cost hinges on the purchasing cost, constraint (22) calculates the holding cost per unit per unit time at time zero ($t = 0$).
- Constraints (23) and (24) along with constraints (25) and (26) find the amounts of the binary variables that are used to define the M_i and C_i in Table 1.
- Constraint (27) shows that we have a limited warehouse space for storage.
- $Z_i = \text{Max} \left\{ 0, \left(\frac{p_i Q_i - b_i}{D_i} - \sum_{j=1}^{m+1} y_{ij} M_{ij} \right) \gamma_i \right\}$ is replaced with the two constraints $Z_i \geq \left(\frac{p_i Q_i - b_i}{D_i} - \sum_{j=1}^{m+1} y_{ij} M_{ij} \right) \gamma_i$ and $Z_i \geq 0$.
- s_i can be redefined as

$$s_i = \begin{cases} 1; & \text{If product } i \text{ benefits discount } (Z_i = 0) \\ 0; & \text{Otherwise } (Z_i > 0) \end{cases}$$

where it is replaced by the constraints $Z_i \times s_i = 0$ and $Z_i + s_i > 0$.

4. Solution algorithms

Since the model of the problem is a constrained nonlinear-programming and hard to be solved by an exact method, a hybrid meta-heuristic algorithm of GA and SA is developed to find a near optimum solution. Besides, a GA is also used to validate the results obtained.

4.1. A genetic algorithm

Besides initial setting of the GA parameters that are population size (N), the elitism percentage (P_e), the crossover percentage (P_c), and the mutation percentage (P_m), this algorithms starts with initialization that involves generating feasible solutions (chromosomes) considering $b_i < Q_i$ in each column of a chromosome. Then it goes to crossover

and mutation of the chromosomes to generate better ones in different iterations. The iterations go on until a predetermined number of generations is reached. Meanwhile, to avoid losing the best chromosomes, in each generation the best chromosomes are saved for the next generation (elitism).

Initial population contains chromosomes that are generated randomly. The GA chromosome of this research is a two by n matrix that shows a solution to the developed inventory model. The first row defines the order quantities and the second indicates the backorder amounts of the products. Figure 2 shows a typical chromosome of the proposed GA.

The chromosomes that are randomly generated may not be feasible under the following two conditions:

- (1) Condition 1: If there is at least a product, whose backorder level is more than its ordered quantity, the corresponding chromosome is infeasible and hence it is replaced with another chromosome.
- (2) Condition 2: A chromosome is also infeasible if the capacity constraint is not satisfied. In this case, the penalty-guided genetic search of David and Smith (1996) is employed. In this case, the total annual cost is defined as:

$$\text{Total cost} = \begin{cases} TC; & F' \leq F \\ TC \times \left(\frac{F}{F'-F}\right); & \text{Otherwise} \end{cases} \quad (32)$$

Note that as a better chromosome is the one with a better objective function value, the objective function value used in the proposed algorithms should be $1/TC$. In this case, a chromosome with a higher fitness is the better one.

In the crossover operation, the roulette wheel selection strategy is first employed to choose the parents and then a matrix of the random crossover mask (Gen & Cheng, 1997) containing zeros and ones is generated. In this paper, instead of a $2 \times n$ matrix, a $1 \times n$ matrix is considered as the random crossover mask. Otherwise, we may face siblings containing columns that the amount of b_i is more than Q_i . Then the crossover operation is performed on the genes of each chromosome column for which their corresponding elements in the random crossover mask matrix is zero. Accordingly, the son and daughter chromosomes are generated from the father and mother chromosomes. Figure 3 depicts an example of this operation.

For the mutation operation, a chromosome from the population is first selected. Then, a linear matrix of size n whose elements are a sample of size n from a uniform distribution in (0,1) is generated randomly. Finally, the mutation operation is performed on the genes of each column of the selected chromosome matrix if their corresponding elements in the random matrix is less than P_m . In this case, the corresponding genes of the selected chromosome are reproduced regarding $b_i > Q_i$ randomly to gain one of its neighbor chromosome. Figure 4 shows an example of the mutation operation.

$$Q_i \begin{bmatrix} Q_1 & Q_2 & \dots & Q_n \\ b_i \begin{bmatrix} b_1 & b_2 & \dots & b_n \end{bmatrix} \end{bmatrix}$$

Figure 2. Chromosome presentation.

$$\begin{array}{l}
 \text{Father : } \begin{array}{l} Q_i \\ b_i \end{array} \begin{bmatrix} 100 & 200 & 120 & 150 \\ 50 & 10 & 28 & 80 \end{bmatrix} \\
 \text{Mother : } \begin{array}{l} Q_i \\ b_i \end{array} \begin{bmatrix} 120 & 300 & 80 & 110 \\ 70 & 56 & 38 & 26 \end{bmatrix} \\
 \\
 \text{Random - crossover - mask : } [1 \ 0 \ 0 \ 1] \\
 \\
 \text{Son : } \begin{array}{l} Q_i \\ b_i \end{array} \begin{bmatrix} 100 & 300 & 80 & 150 \\ 50 & 56 & 38 & 80 \end{bmatrix} \\
 \text{Daughter : } \begin{array}{l} Q_i \\ b_i \end{array} \begin{bmatrix} 120 & 200 & 120 & 110 \\ 70 & 10 & 28 & 26 \end{bmatrix}
 \end{array}$$

Figure 3. An example of the crossover operation.

$$\begin{array}{l}
 \text{Selected - chromosome : } \begin{array}{l} Q_i \\ b_i \end{array} \begin{bmatrix} 100 & 200 & 120 & 150 \\ 50 & 10 & 28 & 80 \end{bmatrix} \\
 \\
 \text{Random - matrix : } [0.8 \ 0.04 \ 0.088 \ 0.78] \\
 \\
 \text{Mutation - offspring : } \begin{array}{l} Q_i \\ b_i \end{array} \begin{bmatrix} 100 & 250 & 120 & 150 \\ 50 & 140 & 28 & 80 \end{bmatrix}
 \end{array}$$

Figure 4. An example of mutation ($P_m = 0.05$).

The algorithm is stopped when a predetermined number of generations are produced.

4.2. The hybrid GA+SA algorithm

The name and the inspiration of the SA algorithm come from the annealing process (a technique involving heating and controlled cooling of a material to increase the size of its crystals to reduce defects) in metallurgy. SA is a process that attempts to move from the current solution to one of its neighbors. It starts from an initial solution and generates a new solution in the neighborhood of the current solution. Then, the change in the objective function value is calculated. If the objective function of the new solution is better then, the new solution is accepted. Otherwise, the transition to the new solution is accepted with a specified probability. The algorithm starts with an initial temperature and in each temperature a special amount of transition occurs. The temperature reduces using a reducing rate until it reaches to the final temperature. Interested readers are referred to Kirkpatrick, Gelatt, and Vecchi (1983) for more details.

The major advantage of SA is its ability to avoid being trapped in local minima. Further, while SA is based on an individual evolution, GA is formed on a population of individuals. Therefore, the combination of these two algorithms may result in better solutions. In this paper, the GA+SA algorithm of Ponnambalam and Reddy (2003) is employed to solve the inventory model at hand.

In the hybrid method, the initial population is first generated by the genetic algorithm. Then, using the GA operators (crossover, mutation, and elitism) the new solutions are produced. Next, each new solution is improved by the SA. After all the solutions of the GA in one generation are exhausted, the best solutions of the population obtained by SA are the solutions of the GA for the next generation. This process continues until a fixed number of generations are generated.

In addition to the parameters of the GA algorithm, the other parameters that are used in the hybrid GA+SA algorithm are the initial temperature (T_0), the final temperature (T_f), and the reducing rate of the temperature (α).

The above two algorithms are coded in Matlab 7.6.0.324 software. Furthermore, an Intel(R) core2 duo CPU 2.00 GHz personal computer with 4 GB RAM is used to solve the numerical problems described later.

5. Tuning the parameters

To validate the results and evaluate the performances of the two proposed algorithms, in this research thirty different examples (in terms of number of products) of random problems are considered in which the number of the products are between 2 and 15. Moreover, in order to tune the parameters of the algorithms, the Taguchi orthogonal arrays are used. The levels of the parameters are shown in Table 2. Using Minitab 15, the results of the Taguchi method are given in Table 3. These tuned parameters have less sensitivity to the number of the products.

Using the calibrated parameters given in Table 3, $30 \times 4 = 120$ randomly generated numerical examples are solved by each algorithm where the results are summarized in Tables 4 and 5. In these tables, each row corresponds to a randomly generated problem and the columns show the four best values in 15 runs along with their estimated standard deviations and their best computer run times. The standard deviation is a good criterion to validate the results obtained by the algorithms in a sense that a smaller standard deviation is an indication of better solution validity.

6. Comparing the algorithms

In this section, the performances of the two proposed algorithms in terms of the mean-fitness and mean-CPU-time are compared using paired t-tests run by Minitab 15. The hypotheses are $\mu_{GA} = \mu_{GA+SA}$ versus $\mu_{GA} \neq \mu_{GA+SA}$ for the two performance measures, resulting in two statistical comparisons. Tables 6 and 7 show the results of the tests using a 95% confidence level.

Table 2. The levels of the parameters in Taguchi method.

Parameters	Levels				
P_c	0.45	0.55	0.65	0.75	0.85
P_m	0.005	0.0162	0.0275	0.0387	0.05
N	75	93.75	112.5	131.25	150
T_0	8	11	14	17	20
T_f	0	1.25	2.5	3.75	5
α	0.7	0.75	0.8	0.85	0.9

Table 3. Tuned parameters of the proposed algorithms.

Algorithm	P_c	P_m	N	T_0	T_f	α
GA	0.75	0.05	150	–	–	–
GA+SA	0.45	0.0162	75	8	2.5	0.9

Downloaded by [Professor Seyed Taghi Akhavan Niaki] at 10:57 08 September 2014

Table 4. The fitness values for different examples solved by GA.

GA	Example number	Fitness ($\times 10^4$)				The best answer	Fitness deviation	Best run time (s)
		Run 1	Run 2	Run 3	Run 4			
Number of products: 2–5	1	6.7169	6.7083	6.7245	6.7225	6.7083	0.01	2.818156
	2	9.3604	9.386	9.3952	9.3615	9.3604	0.02	2.878447
	3	8.6452	8.624	8.6714	8.5885	8.5885	0.04	2.798168
	4	16.064	16.073	16.034	16.101	16.034	0.03	3.254634
	5	18.993	18.866	18.929	18.909	18.866	0.05	3.13979
	6	8.7238	8.7221	8.6963	8.6708	8.6708	0.03	3.395474
	7	15.133	14.971	15.196	15.119	14.971	0.10	3.485488
	8	17.781	17.689	17.568	17.664	17.568	0.09	3.419981
	9	16.96	17.107	17.039	16.999	16.96	0.06	3.639452
	10	18.131	18.059	17.926	17.886	17.886	0.11	3.656244
Number of products: 6–10	11	13.607	13.615	13.687	13.625	13.607	0.04	4.946175
	12	15.39	15.38	15.333	15.36	15.333	0.03	5.325483
	13	13.723	13.806	13.852	13.802	13.723	0.05	4.945532
	14	28.815	28.706	28.923	28.769	28.706	0.09	5.646025
	15	41.105	41.225	41.261	41.143	41.105	0.07	5.834177
	16	28.36	28.378	28.684	28.662	28.36	0.18	5.966434
	17	22.42	22.183	22.153	22.309	22.153	0.12	5.980521
	18	29.98	29.611	30.049	29.491	29.491	0.27	6.11245
	19	33.953	34.431	34.202	34.181	33.953	0.20	6.507385
	20	37.513	37.598	37.538	37.445	37.445	0.06	6.263783
Number of products: 11–15	21	43.569	43.77	43.883	43.784	43.569	0.13	3.268233
	22	39.998	40.079	40.25	40.01	39.998	0.12	3.395825
	23	31.152	31.368	31.139	31.518	31.139	0.18	3.510922
	24	36.79	37.301	37.133	37.247	36.79	0.23	3.767299
	25	35.295	35.249	35.291	35.298	35.249	0.02	3.497581
	26	29.132	29.395	29.425	29.476	29.132	0.15	3.653236
	27	44.461	44.896	44.83	45.124	44.461	0.28	3.760996
	28	28.506	28.628	28.863	28.698	28.506	0.15	3.727078
	29	47.802	47.904	48.076	48.176	47.802	0.17	4.110455
	30	43.439	43.677	43.827	43.673	43.439	0.16	4.139601

Based on the results in Tables 6 and 7, the proposed GA+SA algorithm leads to better solutions in terms of the mean fitness values. However, in terms of the average CPU time, GA+SA takes more time to solve the problems. Moreover, the above tests have been performed using different problems with various sizes (small, medium, and large) separately. The solutions of all problems (not shown here to save spaces) indicate that in small-sized problems, GA+SA results in better quality solutions with a better-solving time. In medium-sized examples, GA takes less time to solve the problems with statistically equal quality. For large-sized problems, while the means of the solving time do not statistically differ from each other, GA+SA results in better quality solutions than GA on average.

Table 5. The fitness values for different examples solved by GA+SA.

GA+SA	Example number	Fitness ($\times 10^4$)				The best answer	Deviation	Best run time (s)
		Run 1	Run 2	Run 3	Run 4			
Number of products: 2–5	1	6.3371	6.3107	6.3752	6.3754	6.3107	0.03	2.056448
	2	8.6756	8.6879	8.725	8.7384	8.6756	0.03	2.028795
	3	8.1343	8.7784	8.2266	8.1093	8.1093	0.31	2.063166
	4	14.871	15.159	14.698	15.136	14.698	0.22	2.323212
	5	18.015	18.287	18.307	18.341	18.015	0.15	2.300851
	6	8.2842	8.1318	8.1451	8.1791	8.1318	0.07	2.289679
	7	14.372	14.345	14.165	14.185	14.165	0.11	2.332317
	8	16.758	16.999	16.728	16.965	16.728	0.14	2.33141
	9	16.222	15.969	15.793	16.314	15.793	0.24	2.580449
	10	17.412	16.844	17.279	17.007	16.844	0.26	2.561804
Number of products: 6–10	11	13.633	13.439	13.274	13.582	13.274	0.16	6.255437
	12	15.054	15.095	15.231	15.261	15.054	0.10	6.253179
	13	13.991	13.575	13.83	13.926	13.575	0.18	6.267034
	14	28.925	29.04	29.024	28.288	28.288	0.36	6.760987
	15	40.777	40.573	40.682	40.066	40.066	0.32	6.86249
	16	27.904	29.337	27.717	28.846	27.717	0.77	7.338948
	17	22.796	21.623	21.936	21.545	21.545	0.57	7.21879
	18	29.32	29.906	30.29	30.246	29.32	0.45	7.282566
	19	34.517	33.444	34.627	34.02	33.444	0.54	7.674959
	20	37.545	36.655	38.254	36.56	36.56	0.80	7.631704
Number of products: 11–15	21	41.645	41.636	41.797	40.916	40.916	0.40	4.345813
	22	37.67	37.344	38.077	38.011	37.344	0.34	4.318904
	23	30.274	29.634	29.588	29.899	29.588	0.31	4.399321
	24	35.102	34.549	34.551	34.824	34.549	0.26	4.558068
	25	33.375	33.997	33.801	33.356	33.356	0.32	4.586358
	26	27.798	27.465	27.211	27.922	27.211	0.32	4.5606
	27	42.109	42.644	41.456	43.635	41.456	0.92	4.761357
	28	27.319	26.716	26.808	27.171	26.716	0.29	4.824383
	29	45.985	45.375	49.954	45.02	45.02	2.28	5.072332
	30	41.135	41.576	40.302	40.54	40.302	0.58	4.573389

Table 6. Fitness paired *t*-test.

	Paired <i>T</i> for GA – GASA			
	<i>N</i>	Mean	SD	SE mean
GA	30	25.99	12.45	2.27
GASA	30	24.76	11.78	2.15
Difference	30	1.227	0.914	0.167

Notes: 95% CI for mean difference: (0.885, 1.568).

T-test of mean difference = 0 (vs. not = 0): *T*-value = 7.35, *p*-value = 0.000.

Downloaded by [Professor Seyed Taghi Akhavan Niaki] at 10:57 08 September 2014

Table 7. Run-time paired *t*-test.

	Paired <i>T</i> for GA – GASA			
	<i>N</i>	Mean	SD	SE mean
GA	30	4.228	1.175	0.215
GASA	30	4.614	1.970	0.360
Difference	30	-0.386	0.990	0.181

Notes: 95% CI for mean difference: (-0.755, -0.016).

T-test of mean difference = 0 (vs. not = 0): *T*-value = -2.13, *p*-value = 0.041.

7. Conclusion and recommendation for future research

In this paper, a multiproduct EOQ problem under permissible delay in payments and discount was studied, in which all the costs increased by an inflation rate at the beginning of the ordering cycles and remained fixed during the cycle. The length of the grace period and the amount of discount depended entirely on the order quantities. The mathematical formulation of the problem was developed and showed hard to solve by an exact method. Accordingly, the model was solved employing two GA and GA+SA meta-heuristic algorithms. The comparison study showed while GA+SA leads to better answers, GA takes less time to solve.

The outline of some recommendations for future research follows.

- (a) Some of the model's parameters such as demand may be considered fuzzy or random.
- (b) Besides the limited warehouse capacity, budget can be another constraint.
- (c) Lot-size deterioration may be added to the problem.

References

- Abad, P. L., & Jaggi, C. K. (2003). A joint approach for setting unit price and the length of the credit period for a seller when end demand is price sensitive. *International Journal of Production Economics*, 83, 115–122.
- Bather, J. A. (1966). A continuous time inventory model. *Journal of Applied Probability*, 3, 538–549.
- Bhaba, R. S., & AL-Kind, M. (2006). Optimal ordering policies in response to a discount offer. *International Journal of Production Economics*, 100, 195–211.
- Borgonovo, E. (2010). Sensitivity analysis with finite changes: An application to modified EOQ models. *European Journal of Operational Research*, 200, 127–138.
- Brander, P., Levén, E., & Segerstedt, A. (2005). Lot sizes in a capacity constrained facility – A simulation study of stationary stochastic demand. *International Journal of Production Economics*, 93–94, 375–386.
- Chao, H. P. (1992). The EQQ model with stochastic demand and discounting. *European Journal of Operational Research*, 59, 434–443.
- David, W. C., & Smith, A. E. (1996). Penalty guided genetic search for reliability design optimization. *Computers & Industrial Engineering*, 30, 895–904.
- Gen, M., & Cheng, R. (1997). *Genetic algorithm and engineering design*. New York, NY: Wiley.
- Goyal, S. K. (1985). Economic order quantity under conditions of permissible delay in payments. *Journal of Operational Research Society*, 36, 35–38.
- Hariga, M. A. (1995). Effects of inflation and time-value of money on an inventory model with time-dependent demand rate and shortages. *European Journal of Operational Research*, 81, 512–520.

- Huang, Y.-F. (2007). Economic order quantity under conditionally permissible delay in payments. *European Journal of Operational Research*, 176, 911–924.
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220, 671–680.
- Leopoldo, E. C. B., Neale, R. S., & Goyal, S. K. (2010). Optimal order size to take advantage of a non-time discount offer with allowed backorders. *Applied Mathematical Modeling*, 34, 1642–1652.
- Liao, H.-C., Tsai, C.-H., & Su, C.-T. (2000). An inventory model with deteriorating items under inflation when a delay in payment is permissible. *International Journal of Production Economics*, 63, 207–214.
- Matsuyama, K. (2001). The EOQ-models modified by introducing discount of purchase price or increase of setup cost. *International Journal of Production Economics*, 73, 83–99.
- Mendoza, A., & Ventura, J. A. (2008). Incorporating quantity discounts to the EOQ model with transportation costs. *International Journal of Production Economics*, 113, 754–765.
- Munson, C. L., & Hu, J. (2010). Incorporating quantity discounts and their inventory impacts into the centralized purchasing decision. *European Journal of Operational Research*, 201, 581–592.
- Ouyang, L.-Y., Teng, J.-T., Goyal, S. K., & Yang, C.-T. (2009). An economic order quantity model for deteriorating items with partially permissible delay in payments linked to order quantity. *European Journal of Operational Research*, 194, 418–431.
- Ponnambalam, S. G., & Reddy, M. M. (2003). A GA-SA multi-objective hybrid search algorithm for integrating lot sizing and sequencing in flow-line scheduling. *International Journal of Advanced Manufacturing Technology*, 21, 126–137.
- Roy, A., & Samanta, G. P. (2011). Inventory model with two rates of production for deteriorating items with permissible delay in payments. *International Journal of Systems Science*, 42, 1375–1386.
- Sana, S. S., & Chaudhuri, K. S. (2008). A deterministic EOQ model with delays in payments and price-discount offers. *European Journal of Operational Research*, 184, 509–533.
- Shinn, S.-W., Hwang, H., & Park, S.-S. (1996). Joint price and lot size determination under conditions of permissible delay in payments and quantity discounts for freight cost. *European Journal of Operational Research*, 91, 528–542.
- Toptal, A. (2009). Replenishment decisions under an all-units discount schedule stepwise freight costs. *European Journal of Operational Research*, 198, 504–510.