Int. J. Production Economics 130 (2011) 77–86

Contents lists available at ScienceDirect

Int. J. Production Economics

journal homepage: www.elsevier.com/locate/ijpe

Multi-objective models for lot-sizing with supplier selection

Jafar Rezaei [a,*], Mansoor Davoodi [b]

[a] Department of Technology, Strategy and Entrepreneurship, Faculty of Technology, Policy and Management, Delft University of Technology, P.O. Box 5015, 2600 GA Delft, The Netherlands

[b] Laboratory of Algorithms and Computational Geometry, Department of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran

ARTICLE INFO

Article history:
Received 24 January 2009
Accepted 18 November 2010
Available online 24 November 2010

Keywords:
Lot-sizing
Supplier selection
Inventory
Multi-objective mixed integer non-linear programming
Genetic algorithm

ABSTRACT

In this paper, two multi-objective mixed integer non-linear models are developed for multi-period lot-sizing problems involving multiple products and multiple suppliers. Each model is constructed on the basis of three objective functions (cost, quality and service level) and a set of constraints. The total costs consist of purchasing, ordering, holding (and backordering) and transportation costs. Ordering cost is seen as an 'ordering frequency'-dependent function, whereas total quality and service level are seen as time-dependent functions. The first model represents this problem in situations where shortage is not allowed while in the second model, all the demand during the stock-out period is backordered. Considering the complexity of these models on the one hand, and the ability of genetic algorithms to obtain a set of Pareto-optimal solutions, we apply a genetic algorithm in an innovative approach to solve the models. Comparison results indicate that, in a backordering situation, buyers are better able to optimize their objectives compared to situations where there is no shortage. If we take ordering frequency into account, the total costs are reduced significantly.

1. Introduction

Recently, the 'Dynamic version of the economic lot size model' paper by Wagner and Whitin (1958) was elected one of the ten most influential publications in the first half century of Management Science (Wagner, 2004), which is an indication of the importance of lot-sizing problem in the area of management science. The authors investigated a single product, multi-period lot-sizing model. In later decades, this problem was extended in several directions. In a comprehensive literature review, Karimi et al. (2003) represented a number of important characteristics of lot-sizing models, including the planning horizon (long term versus short term), number of levels (single level versus multi-level), number of products (single item versus multiple items), capacity or resource constraints (capacitated versus incapacitated), deterioration of items, demand, setup structure and shortage. Interested readers are referred to Robinson et al. (2009), Ben-Daya et al. (2008) and Karimi et al. (2003) for different models and classifications of the lot-sizing problem.

One recent approach to this problem, examining the increasing importance of supply chain management (SCM), takes a combined look at lot-sizing and supplier selection. Few papers in this area (e.g. Rezaei and Davoodi, 2008; Dai and Qi, 2007; Basnet and Leung, 2005) discuss situation where buyers can simultaneously select the most suitable suppliers for each period and optimize the lot size of each product. The implicit assumption of these papers is an arm's length relationship between buyer and supplier, as they merely emphasize the role of costs in the decision-making process, ignoring the potential role of other essential factors in facilitating cooperation (Mentzer et al., 2001; Morgan and Hunt, 1994), which means that in these papers, the problem has been formulated in a single objective format based on purchasing cost, holding cost and ordering cost.

Although transportation costs form a substantial part of the total logistics costs of a product, they are surprisingly often ignored in the bulk of lot-sizing research (van Norden and van de Velde, 2005). Ertogral et al. (2007) explicitly integrated the transportation cost in the single-vendor single-buyer problem and concluded that this combination can reduce the overall costs of a system.

In this paper, we combine the lot-sizing problem with supplier selection and present two multi-objective models with regard to shortage occurrence. Although, in the framework of SCM, where there is some agreement between buyer and supplier, for instance in the form of a *procurement collaboration* (Meyr et al., 2008), it is possible to avoid shortage, it sometimes is inevitable or planned (Sharafali and Co, 2000). Large companies that have successfully implemented SCM, like Dell, Cisco Systems and HP, are sometimes faced with inventory shortage (Walsh, 2010; Gollner, 2008). There are several causes for inventory shortage, including part variations, mis-operation, inventory reduction (Jiang et al., 2010), small number of suppliers, conservative production plan of suppliers (Xu, 2010) and supplier's service level. As such, the problem should

* Corresponding author. Tel.: +31 15 27 82985; fax: +31 15 27 83177.
E-mail addresses: j.rezaei@tudelft.nl (J. Rezaei), mdmonfared@aut.ac.ir (M. Davoodi).

0925-5273/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.ijpe.2010.11.017

be studied in two different situations: (1) when shortage is avoidable and (2) when shortage is inevitable or planned. Taking these two different scenarios into account, the first model is formulated assuming the shortage is not allowed, whereas in the second model the shortage is assumed to be allowed and backordered.

Thus far, only a few researchers have used genetic algorithms to try and solve the inventory and lot-sizing problems in general (Gupta et al., 2007; Rezaei and Davoodi, 2005; van Hop and Tabucanon, 2005; Dellaert et al., 2000; Disney et al., 2000) and 'lot-sizing with supplier selection problem' in particular (e.g. Sadeghi Moghadam et al., 2008; Rezaei and Davoodi, 2008, 2006; Liao and Rittscher, 2007; Xie and Dong, 2002; Dellaert et al., 2000). In this paper, we look at some robust and unique characteristics of genetic algorithms, especially those dealing with multi-objective problems, and adopt a genetic algorithm approach to solve these problems, introducing a new flexible approach to dealing with hard constraints in multi-objective optimization problems.

The remainder of this paper is organized as follows. In Section 2, we present the formulation of the models. In Section 3, we present a genetic algorithm to solve these models. In Section 4, two numerical examples and comparison results are presented. Finally, in Section 5, we provide our conclusions and offer suggestions for future research.

2. Mathematical modeling

In this section, we present two multi-objective mixed-integer non-linear programming (MOMINLP) models: (1) an MOMINLP model without shortage and (2) an MOMINLP model with backordering, both of which with three objective functions and a set of constraints. We use the following notations to formulate the models.

Notations

I	number of products
J	number of suppliers
T	number of periods
x_{ijt}	number of product i ordered to supplier j in period t
p_{ij}	net purchase cost of product i from supplier j
o_j	ordering cost for supplier j
g_j	transportation cost for supplier j per vehicle
y_{jt}	binary integer: 1, if the order is given to supplier j in period t, 0, otherwise
h_i	holding cost of product i per period
d_{it}	demand of product i in period t
f_{ijt}	quality level of product i offered by supplier j in period t
s_{ijt}	service level of product i offered by supplier j in period t
λ_{ij}	quality level growth rate of product i offered by supplier j
γ_{ij}	service level growth rate of product i offered by supplier j
β_j	ordering cost reduction rate for supplier j
c_{ij}	capacity of supplier j in production of product i per period
b_i	backordering cost of product i
w_i	occupied space by product i in warehouse or vehicle
W	total storage capacity
v_j	vehicle capacity for supplier j

2.1. MOMINLP model without shortage

In this model, there are three objectives: total cost, total quality level and total service level, and a set of constraints. The problem is to determine which products to order in which quantities from which suppliers in which periods, in order to satisfy overall demand. The main assumption is that shortage is not allowed. In the following section, we describe the components and formulation of the problem.

2.1.1. Objective functions

Total cost: The sum of the purchasing costs, ordering costs, holding costs and transportation costs in all periods should be minimized. Most existing studies only include the first three types of costs and ignore transportation costs. The total purchasing costs are the sum of the purchasing costs of all products from all selected suppliers in all periods. In most cases, ordering costs are formulated as $o_j y_{jt}$ where o_j is the ordering cost for supplier j and y_{jt} a binary variable that is 1 if an order is placed with supplier j in time period t and that otherwise is 0. However, in most real situations, this is not the case, especially in an SCM framework. As pointed out in many studies (e.g. Spekman et al., 1998; Lambert, 2008), the key driver of buyers in SCM is cost reduction. As Woo et al. (2001) have found, reduction in ordering cost is positively related to ordering frequency, in other words, the higher the ordering frequency, the higher the ordering cost reduction, which is why we propose an exponential relationship between the total ordering cost for supplier j and the number of orders (order frequency) placed with that supplier ($\sum_{k=1}^{t} y_{jk}$), which results in the following cost formula: $o_j e^{-\beta_j \sum_{k=1}^{t} y_{jk}} y_{jt}$. In formulating the holding costs, we should note that, because the supplier's service level is not necessarily 100%, the total number of received items in period t is not necessarily equal to the total number of ordered items in the same period.

With regard to transportation costs, we assume that the buyer, based on criteria like geographic distance to the supplier, uses different vehicles with different capacities. However, these vehicles can be used for all kinds of ordered products.

Therefore we have:

$$\min z_1 = \sum_i \sum_j \sum_t p_{ij} x_{ijt} + \sum_j \sum_t o_j e^{-\beta_j \sum_{k=1}^{t} y_{jk}} y_{jt} + \sum_i \sum_t h_i \left(\sum_{k=1}^{t} \sum_j x_{ijk} - \sum_j (1 - s_{ij0} e^{\gamma_{ij} t}) x_{ijt} - \sum_{k=1}^{t} d_{ik} \right) + \sum_j \sum_t g_j \left\lceil \frac{\sum_i w_i x_{ijt}}{v_j} \right\rceil \quad (1)$$

Total quality level: Product quality is defined in terms of conformance to product-related customer requirements, where customer requirements are the requirements that should be fulfilled to meet the customer's needs (Berden et al., 2000). The overall quality level of all the products ordered from all suppliers in all periods should be maximized. In existing literature (e.g. Liao and Rittscher, 2007; Amid et al., 2006), the quality level of products is implicitly assumed to be equal in all periods. However, product quality may vary over time. In this paper, we use an exponential time-dependent function for the quality level of individual products as $f_{ij0} e^{\lambda_{ij} t}$, where f_{ij0} is the quality level of product i offered by supplier j at the first point of planning horizon. If the buyer predicts that the quality level of product i offered by supplier j will have an increasing trend $\lambda_{ij} > 0$, otherwise $\lambda_{ij} \le 0$. This results in the following formula:

$$\max z_2 = \sum_i \sum_j \sum_t f_{ij0} e^{\lambda_{ij} t} x_{ijt} \quad (2)$$

Total service level: We adopt the β service level from Schneider (1981) and, for the purpose of this paper, define the service level of supplier j for product i in period t as follows:

$$s_{ijt} = \frac{\text{satisfied ordered items of product } i \text{ to the supplier } j \text{ in period } t}{\text{ordered items of product } i \text{ to the supplier } j \text{ in period } t}$$

Because the supplier's service level affects the buyer, especially with regard to safety stock levels and costs, the buyer wants to maximize the overall service level of all products ordered from all suppliers in all periods. The construction of this function is similar to the previous objective function, while it is different from existing studies literature in terms of its predictability, which is why we use an exponential time-dependent function

$$\max z_3 = \sum_i \sum_j \sum_t s_{ij0} e^{\gamma_{ij} t} x_{ijt} \tag{3}$$

It is assumed that undelivered items that were ordered in period t will be delivered by the end of the next period.

2.1.2. Constraints

The constraints of this problem are formulated as follows:

Demand: This constraint stipulates that all requirements must be met in the period in which they originate. In other words, shortage or backordering is not allowed. As a result, because the supplier's service level is not necessarily 100%, the buyer should maintain a safety stock, which for product i in period t is calculated as $\sum_j (1-s_{ij0} e^{\gamma_{ij} t}) x_{ijt}$, resulting in the following formula:

$$\sum_{k=1}^{t} \sum_j x_{ijk} - \sum_{k=1}^{t} d_{ik} \ge \sum_j (1-s_{ij0} e^{\gamma_{ij} t}) x_{ijt} \quad \text{for all } i \text{ and } t \tag{4}$$

Charging ordering cost: According to this constraint, buyers cannot place an order without facing appropriate ordering costs. Note that there is no any contradiction between this constraint and the above-mentioned formulation of ordering costs in z_1.

$$\left(\sum_{k=t}^{T} d_{ik} \right) y_{jt} - x_{ijt} \ge 0 \quad \text{for all } i, j \text{ and } t \tag{5}$$

Inventory at the end of planning horizon: This constraint guarantees that, at the end of planning horizon, the inventory level of each item will be zero

$$\sum_{t=1}^{T} \sum_j x_{ijt} - \sum_j (1-s_{ij0} e^{\gamma_{ij} T}) x_{ijT} - \sum_{t=1}^{T} d_{it} = 0 \quad \text{for all } i \tag{6}$$

However, if the buyer wants to hold inventory at the end of planning horizon, we can replace Eq. (6) with $\sum_{t=1}^{T} \sum_j x_{ijt} - \sum_j (1-s_{ij0} e^{\gamma_{ij} T}) x_{ijT} - \sum_{t=1}^{T} d_{it} \ge \psi_i$ for all i, where ψ_i indicates the lower bound of inventory of product i at the end of planning horizon.

Storage capacity: This constraint shows that the buyer has a limited storage capacity in each period:

$$\sum_i w_i \left(\sum_{k=1}^{t} \sum_j x_{ijk} - \sum_j (1-s_{ij0} e^{\gamma_{ij} t}) x_{ijt} - \sum_{k=1}^{t} d_{ik} \right) \le W \quad \text{for all } t \tag{7}$$

Supplier capacity: This constraint indicates that the number of products i ordered from supplier j in period t should be equal to or less than the supplier's capacity to deliver this product.

$$x_{ijt} \le c_{ij} \quad \text{for all } i, j \text{ and } t \tag{8}$$

Binary and non-negativity constraints:

$$y_{jt} = 0 \text{ or } 1 \quad \text{for all } j \text{ and } t, \quad x_{ijt} \ge 0 \quad \text{for all } i, j \text{ and } t \tag{9}$$

In this problem, because shortage is not allowed and suppliers have limited capacity, the following inequality has to be satisfied:

$$\sum_{k=1}^{t} d_{it} \le \sum_j t c_{ij} \quad \text{for all } i \text{ and } t \tag{10}$$

The resulting multi-objective mixed integer non-linear programming (MOMINLP) model looks as follows:

$$\min z_1 = \sum_i \sum_j \sum_t p_{ij} x_{ijt} + \sum_j \sum_t o_j e^{-\beta_j \sum_{k=1}^{t} y_{jk}} y_{jt} + \sum_i \sum_t h_i \left(\sum_{k=1}^{t} \sum_j x_{ijk} - \sum_j (1-s_{ij0} e^{\gamma_{ij} t}) x_{ijt} - \sum_{k=1}^{t} d_{ik} \right) + \sum_j \sum_t g_j \left\lceil \frac{\sum_i w_i x_{ijt}}{v_j} \right\rceil \tag{11}$$

$$\max z_2 = \sum_i \sum_j \sum_t f_{ij0} e^{\lambda_{ij} t} x_{ijt} \tag{12}$$

$$\max z_3 = \sum_i \sum_j \sum_t s_{ij0} e^{\gamma_{ij} t} x_{ijt} \tag{13}$$

subject to

$$\sum_{k=1}^{t} \sum_j x_{ijk} - \sum_{k=1}^{t} d_{ik} \ge \sum_j (1-s_{ij0} e^{\gamma_{ij} t}) x_{ijt} \quad \text{for all } i \text{ and } t \tag{14}$$

$$\left(\sum_{k=t}^{T} d_{ik} \right) y_{jt} - x_{ijt} \ge 0 \quad \text{for all } i, j \text{ and } t \tag{15}$$

$$\sum_{t=1}^{T} \sum_j x_{ijt} - \sum_j (1-s_{ij0} e^{\gamma_{ij} T}) x_{ijT} - \sum_{t=1}^{T} d_{it} = 0 \quad \text{for all } i \tag{16}$$

$$\sum_i w_i \left(\sum_{k=1}^{t} \sum_j x_{ijk} - \sum_j (1-s_{ij0} e^{\gamma_{ij} t}) x_{ijt} - \sum_{k=1}^{t} d_{ik} \right) \le W \quad \text{for all } t \tag{17}$$

$$x_{ijt} \le c_{ij} \quad \text{for all } i, j \text{ and } t \tag{18}$$

$$y_{jt} = 0 \text{ or } 1 \quad \text{for all } j \text{ and } t, \quad x_{ijt} \ge 0 \quad \text{for all } i, j \text{ and } t \tag{19}$$

2.2. MOMINLP model with backorder

In the previous model, we assumed that shortage is not allowed. In the second model, we examine a situation in which shortage will not immediately result in a lost sale, but shortage and backordering are allowed. However, the backorder not only takes customer service time to answer the inquiry, there are also costs involved in, for instance, shipping the product once it arrives at the distribution center and there are usually intangible costs. These shortage-related costs apply per unit of product i per period as b_i. As it is beyond the scope of this paper to measure the backordering costs, we refer interested readers to Liberopoulos et al. (2010), Anderson et al. (2006) and Centikaya and Parlar (1998), among others.

The resulting model in this situation is

$$\min z_1 = \sum_i \sum_j \sum_t p_{ij} x_{ijt} + \sum_j \sum_t o_j e^{-\beta_j \sum_{k=1}^{t} y_{jk}} y_{jt} + \sum_i \sum_t h_i \left(\sum_{k=1}^{t-1} \sum_j x_{ijk} + \sum_j s_{ij0} e^{\gamma_{ij} t} x_{ijt} - \sum_{k=1}^{t} d_{ik} \right)^{+} - \sum_i \sum_t b_i \left(\sum_{k=1}^{t-1} \sum_j x_{ijk} + \sum_j s_{ij0} e^{\gamma_{ij} t} x_{ijt} - \sum_{k=1}^{t} d_{ik} \right)^{-} + \sum_j \sum_t g_j \left\lceil \frac{\sum_i w_i x_{ijt}}{v_j} \right\rceil \tag{20}$$

$$\max z_2 = \sum_i \sum_j \sum_t f_{ij0} e^{\lambda_{ij} t} x_{ijt} \tag{21}$$

$$\max z_3 = \sum_i \sum_j \sum_t s_{ij0} e^{\gamma_{ij} t} x_{ijt} \tag{22}$$

subject to

$$\left(\sum_{k=t}^{T} d_{ik} \right) y_{jt} - x_{ijt} \ge 0 \quad \text{for all } i, j \text{ and } t \tag{23}$$

$$\sum_i w_i \left(\sum_{k=1}^{t-1} \sum_j x_{ijk} + \sum_j s_{ij0} e^{\gamma_{ij} t} x_{ijt} - \sum_{k=1}^{t} d_{ik} \right)^{+} \le W \quad \text{for all } t \tag{24}$$

$$\sum_{t=1}^{T} \sum_j x_{ijt} - \sum_j (1 - s_{ij0} e^{\gamma_{ij} T}) x_{ijT} - \sum_{t=1}^{T} d_{it} = 0 \quad \text{for all } i \tag{25}$$

$$0 \le x_{ijt} \le c_{ij} \quad \text{for all } i,\ j \text{ and } t \tag{26}$$

$$y_{jt} = 0 \text{ or } 1 \quad \text{for all } j \text{ and } t \tag{27}$$

With regard to the complexity of the ranger of capacitated lot-sizing problems presented by Bitran and Yanasse (1982) and Florian et al. (1980), the models presented here are also NP-hard.

In addition, we are dealing here with a multi-objective type of the problem that contains two non-linear functions. Due to the complexity of models, we use the NSGA-II algorithm (Deb et al., 2000), with some innovations dealing with hard constraints.

3. Genetic algorithm

Since, in multi-objective problems, the objectives are usually conflicting, the first goal in solving a multi-objective problem is finding Pareto-optimal solutions. A Pareto-optimal solution is a solution that is not improved but is only possible by sacrificing one objective to maximize another. The second goal is finding a set of Pareto solutions as diverse as possible in an objective space. These two goals are orthogonal to each other, while there are some algorithms that satisfy only one of them (Deb, 2001). Because classic search and optimization methods use a point-by-point approach, they obtain a single optimal solution, but because in the Evolutionary Algorithms (*EAs*) a population of solutions is applied in each iteration, *EAs* are more suitable when it comes to achieving both objectives of *MOO* problems (optimality and diversity).

Genetic Algorithms (GAs) are among the most robust probabilistic search techniques with regard to large, complex, non-convex, discrete search space or ill-behaved objective functions (Deb, 2001; Goldberg, 1989). They are among the population-based heuristic search algorithms that start with an initial set of (so-called individual or offspring) solutions which are then evolved toward better solutions via certain genetic operators. In the last two decades, many modifications and optimizations have been developed regarding the convergence rate and quality of final solutions of GAs (cf. Michalewicz, 1994; Goldberg, 1989). Most of these variations refer to MOO problems (Augusto et al., 2006; Zitzler et al., 2001; Srinivas and Deb, 1994) and focus on the goals of MOO, resulting in a number of powerful MOO algorithms, such as NSGA-II and SPEA2, which are evolved versions of NSGA and SPEA, respectively (for more information on this subject, see Deb, 2001).

To solve both models presented in this paper, we use NSGA-II, a robust MOO genetic algorithm, which has proposed by Deb et al. (2000). The algorithm uses a fixed-sized population and divides that population into various subpopulations that are different on the preferred levels. It should be said that solution q is dominated by solution p if and only if p is better than q with regard to all the objectives, or p is better than q with regard to at least one objective and not worse than q with regard to other objectives. This is called the non-dominated principle. Each solution in the ith subpopulation denoted by F_i is dominated by at least one solution in the high level subpopulations like F_j where $j < i$. NSGA-II uses a globally elitist strategy and tries to satisfy the second goal of MOEAs simultaneously via convergence to Pareto-optimal solutions. The NSGA-II procedure is outlined as follows (Deb, 2001):

3.1. NSGA-II algorithm

Step 0: Set $t=0$. Create initial population and call it P_t. Use selection operator and genetic operators and generate offspring population, Q_t.

Step 1: Set $R_t = P_t \cup Q_t$. Perform a non-dominated sorting to R_t and identify different fronts: F_i, $i = 1, 2, \ldots$, etc.

Step 2: Set new population $P_{t+1} = \emptyset$ and $i = 1$.

Until $|P_{t+1}| + |F_i| < N$ perform $P_{t+1} = P_{t+1} \cup F_i$ and $i = i+1$.

Step 3: For remaining capacity in P_{t+1}, perform the crowding operators and fill it by some of the best solutions in F_i.

Step 4: Generate offspring population Q_{t+1} from P_{t+1} by using selection and genetic operators.

N is the size of the (parent and offspring) populations. In the following section, we describe our *GA* operators and other aspects to solve the models presented earlier. Also, to increase the power of the algorithm to satisfy the models constraint, we propose a new operator called refiner operator.

3.2. Encoding

The encoding process establishes the way the solutions are represented in chromosomes. We take each chromosome as a model solution, where *I*, *J* and *T* are the number of products, suppliers and periods, respectively, and each chromosome is an integer vector *X* by length $I \times J \times T$ and a binary vector *Y* by length $J \times T$, appropriate by each x_{ijt} and y_{jt} (see Fig. 1).

3.3. Initialize of population

Since, in the complex space, especially in *MOO* problems, there is no information about the location of optimal regions in a search landscape, in order to create initial chromosomes, the most universal method is to randomly generate solutions for the whole population in a uniform way. We use a random binary generator for *Y*, and a random integer generator for *X* with respect to the bounding conditions for all x_{ijt} (constraints 18 and 26). Also, we maintain this satisfaction in the GA process (after crossover and mutation operators).

3.4. Selection strategy

In this paragraph, we explain three elements of GA in this paper: fitness function, selection operator and model constraint satisfaction.

In each model, there are three objectives. The first model has four constraints (14–17), while the second has three (23–25) (please note that we have not counted the boundary and binary constraints here). We use constraints and objective functions to evaluate the fitness of each chromosome. Some approaches have been suggested to satisfy the constraints of MOO problems, including ignoring infeasible chromosomes, penalty function, JVGS, constrained tournament, etc. (Deb, 2001; Homaifar et al., 1994; Michalewicz and Schoenauer, 1996; Jimenez et al., 1999). We use the constrained tournament method because of its ability to satisfy constraints and carry out selection based on fitness at the same time (unlike other approaches, like roulette wheel and proportionate selection). Indeed, the constrained tournament selection

x_{IJT}	x_{IJT-1}	…	x_{ijt}	…	x_{112}	x_{111}

y_{JT}	y_{JT-1}	…	y_{jt}	…	y_{12}	y_{11}

Fig. 1. Each chromosome consists of two integers and 0/1 vectors.

operator combines domination principle and constraint satisfaction.

This operator first selects two (or more) chromosomes from a population and then selects a winner in a tournament with the following rules:

- A feasible solution is better than an infeasible solution.
- Between two feasible solutions (or two infeasible solutions), the standard domination identifies the winner.

In the second rule, the constrained tournament selection operator first attends to objective functions (first goal of MOO) and then to the diversity of the population (second goal of MOO). Several approaches have been suggested to achieve diversity, like niche metrics, crowding models, sharing functions, etc. (Deb, 2001). In this paper, we use crowding distance based on average distance of solution in objective space.

Since constraints 16 and 25 in the models are *hard constraints* (constraints that need to satisfy an equation), we use a new, simple and flexible method to satisfy them. Our strategy for dealing with these constraints in the models is that we consider a specific percentage of the chromosomes population with low constraints violation (only with regards to constraints 16 and 25) feasible solutions for these constraints (though not for all constraints). This flexibility allows the solutions of the first generations of algorithm to be alive in the population. The level of flexibility is identified with regard to the generation number. In other words, it is reduced while the generation number is increased and after some generations it is set to the zero level and leads to the true feasible solutions. If *sol* is a solution in population and we want to assign a value to its violation constraints in generation number *gn* (which is done for all types of hard constraints), we begin by computing the amount of violation of *sol*, we have to satisfy $g_i(sol)=b_i$, after which $VC_i(sol)=|g_i(sol)-b_i|$, and we use the following relationship:

$$HC_i(sol)=\begin{cases}0 & \text{if } gn<fn \text{ and } VC_i(sol)<\alpha e^{1/gn}\\ VC_i(sol) & \text{otherwise}\end{cases} \tag{28}$$

where $HC_i(sol)$ is the value of the violation constraint in hard constraint, *fn* the maximum number of flexible generations (flexible level) and α a sufficient coefficient. After computing the HC_i (it is equal to VC_i after the *gn*th generation), the constrained tournament operator uses this instead of VC_i.

fn and α are identified with regard to the termination condition (maximum number of generations) and the average VC_i population.

3.5. Genetic operators

Genetic operators (reproduction operators) are used to generate new offspring and develop the next population from the current population. Generally speaking, there are two types of genetic operators: crossover and mutation operators.

Crossover operator: Crossover operators combine information from two (or more) parents (solutions of the current population) in such a way that the two (or more) children (solutions for the next population) resemble each parent. There are several available methods to do so (Deb, 2001; Michalewicz, 1994). Since, in this study, each chromosome contains two vectors, integer (X) and binary (Y), we use a single point linear crossover to generate an integer vector, and a single point multiple replacement to generate a binary vector. The following pseudo-code describes the crossover operation:

$\lambda=$ random real value between 0 and Max_λ

$cpx=$ a random integer number between 0 and $I\times J\times T$

$$\begin{cases}\text{for all } i,\ j,\ t \text{ do following operation:} & \\ C_1(x_{ijt})=\lambda P_1(x_{ijt})+(1-\lambda)P_2(x_{ijt}) & \text{if } i\times j\times t<cpx\\ C_2(x_{ijt})=\lambda P_2(x_{ijt})+(1-\lambda)P_1(x_{ijt}) & \text{otherwise}\end{cases}$$

$cpy=$ a random integer number between 0 and $J\times T$

$$\begin{cases}\text{for all } j,\ t \text{ do following operation:} & \\ C_1(y_{jt})=P_1(y_{jt}) & \text{if } j\times t<cpy\\ C_1(y_{jt})=P_2(y_{jt}) & \text{otherwise}\\ C_2(y_{jt})=P_2(y_{jt}) & \text{if } j\times t<cpy\\ C_2(y_{jt})=P_1(y_{jt}) & \text{otherwise}\end{cases}$$

where P_1 and P_2 are two selected parents and where C_1 and C_2 are two children. After conducting the linear crossover, we first round the result (because X is an integer vector) and check its lower and upper bounds (constraints 16 and 25); this type of violation takes place for a Max_λ with a value greater than one.

Mutation operator: Mutation operators alter or mutate one chromosome by changing one or more variables in some way or by some random amount to form one offspring. Here, we use a linear mutation by probability $1/I\times J\times T$ for mutating X vector and bit-wise mutation by probability $1/J\times T$ for Y vector. For mutating X, we first select a random cell x_{ijt} and then, with regard to the upper and lower bounds, select a new integer at random for X and for Y and replace its value by 1 if it is 0 and vice versa.

3.6. Refiner operator

Despite the *objectives space* in the models presented in this paper, which is a three-dimension space, the *search space* is a very big space with ($IJT+JT$) dimensions. With regard to the upper-bound integer decision variable of the problem (see Eqs. (16) and (25)) and the binary variables of the models, it can be said that the number of solutions in the search space is $2^{JT}\prod_{i=1}^{I}\prod_{j=1}^{J}\prod_{t=1}^{T}c_{jt}$, while taking the constraints of the models into account, only a few number of them are feasible. In both models, there are two or three soft constraints and one hard constraint. Our initial approach to satisfying these constraints is via the flexibility violation level presented in Eq. (28). Here, we propose a new operator called *refiner operator*, which after generating a solution tries to guide that solution towards the feasible region, with the operator computing $s=(\sum_{k=1}^{T}d_{jt})y_{jt}-x_{ijt}$ for a fixed value i, j and t. If $s<0$, the operator reduces the value of x_{ijt}, especially if $y_{it}=0$ it sets x_{ijt} to zero for all i, too. On the other hand, if $s\geq 0$, the constraint (15 or 23) is satisfied but, usually, satisfying the other two constraints is difficult because they usually contradict each other. So, the refiner operator increases the value of x_{ijt}.

Furthermore, to satisfy the hard constraint, the operator computes $s=\sum_{t=1}^{T}\sum_j x_{ijt}-\sum_j(1-s_{ij0}e^{\gamma_{ij}T})x_{ijT}-\sum_{t=1}^{T}d_{it}$ for a fixed value i. If $s<0$, the operator increases the value of some x_{ijt} for $0<t<T$. Otherwise, if $s>0$, the value of x_{ijt}, which is selected randomly for some t where $0<t<T$, is increased. In all cases, the increases or reductions have a random value in [0, ABS(s)].

3.7. Termination condition

The selection, crossover, mutation and refiner operators are repeated until a termination condition has been reached. In the single objective GA, there are several conditions for terminating the *GA* process, for instance when a solution is found that satisfies the minimum criteria, when a fixed number of generations has been reaching, when the allocated budget (computation time/money) is reached, etc. In MOO problems, because of the specific objectives (obtaining a set of diverse Pareto-optimal solutions) only some of these criteria, for instance the maximum number of generations, can be used. However, identifying an exact and efficient number of iterations is a difficult and empirical task, it can be determined with regard to the size of population, complexity of search space and number of final non-dominated solutions. In the simulation presented in this paper, we use a population size of 100, 20 final non-dominated solutions and a maximum of 500 iterations.

4. Numerical examples

In this section, we apply the GA discussed earlier to solve two examples to illustrate the proposed models and compare the results. The data used in these examples consists of three products, five suppliers and four periods, resulting in 60 decision-making variables x_{ijt}. The results indicate which products to order in which quantities from which suppliers in which periods (see Tables 1–3).

4.1. Results (model 1)

In this paragraph, we discuss the results obtained from model 1. As mentioned before, the designed genetic algorithm provides us with a set of 20 non-dominated solutions. In Table 4, we find the value of objectives functions z_1, z_2 and z_3 for this set of solutions, while Table 5 illustrates only three members of this set (Sol.15 indicates the value of the decision-making variables of a non-dominated solution with the lowest value of z_1, Sol.2 shows the value of decision-making variables of a non-dominated solution with the highest value of z_2 and Sol.4 exhibits the value of decision-making variables of a non-dominated solution with the highest value of z_3).

Figs. 2–4 show the objective trade-off between total costs and total quality level, total costs and total service level and total

Table 1
Demand for three products in four periods.

	Product (i)	Period (t)			
		1	2	3	4
d_{it}	1	454	540	675	755
	2	327	320	290	285
	3	645	650	637	663

Table 2
Holding costs, backordering costs and occupied space of each product.

Product (i)	h_i	b_i	w_i
1	12	17	0.75
2	35	38	0.85
3	7	10	0.60

Table 3
Other data related to products and/or suppliers.

	Product (i)	Supplier (j)				
		1	2	3	4	5
p_{ij}	1	66	73	62	60	71
	2	121	134	127	142	122
	3	28	26	29	25	23
f_{ij0}	1	0.95	0.98	0.92	0.96	0.99
	2	0.89	0.92	0.95	0.90	0.91
	3	0.93	0.87	0.88	0.83	0.87
s_{ij0}	1	0.95	0.97	0.94	0.99	0.98
	2	0.92	0.99	0.91	0.99	0.92
	3	0.99	0.98	0.99	0.97	0.99
c_{ij}	1	550	480	420	890	790
	2	570	85	450	235	525
	3	825	430	850	390	720
λ_{ij}	1	0.001	−0.002	0.0015	−0.001	−0.001
	2	0.00	0.0015	−0.001	0.002	0.002
	3	−0.001	0.002	0.0015	0.0015	0.0012
γ_{ij}	1	0.0013	0.0011	−0.001	0.001	−0.0011
	2	0.00	0.0014	0.002	0.0015	0.002
	3	−0.0011	0.0016	−0.001	−0.002	0.0011
β_j		0.08	0.1	0.09	0.12	0.09
o_j		45 000	64 500	37 250	53 400	42 250
v_j		85	150	85	100	150
g_j		33 500	55 200	33 500	38 400	55 200

$W=150$

Table 4
Objective functions values of non-dominated solutions (model 1).

Solution no.	z_1	z_2	z_3
Sol.1	3116 505	5804.954	6107.61
Sol.2	**3228 229**	**5847.146**	**6113.339**
Sol.3	3190 892	5825.799	6117.82
Sol.4	**3235 358**	**5777.225**	**6120.463**
Sol.5	3192 673	5770.626	6119.889
Sol.6	3018 613	5786.745	6076.588
Sol.7	3029 292	5786.011	6078.704
Sol.8	3062 160	5784.508	6079.36
Sol.9	3076 930	5785.23	6079.48
Sol.10	3017 829	5785.734	6077.947
Sol.11	3171 162	5767.763	6119.976
Sol.12	3098 473	5767.924	6118.047
Sol.13	3169 479	5767.074	6118.72
Sol.14	3087 668	5808.264	6102.401
Sol.15	**3013 904**	**5786.101**	**6076.555**
Sol.16	3077 198	5786.569	6078.527
Sol.17	3077 453	5785.612	6078.839
Sol.18	3050 891	5787.11	6077.786
Sol.19	3084 527	5787.152	6077.787
Sol.20	3085 014	5787.635	6077.745

Table 5
Decision-making variables value of three non-dominated solutions (model 1).

x_{ijt}	Sol.2	Sol.4	Sol.15	x_{ijt}	Sol.2	Sol.4	Sol.15
x_{111}	158	105	0	x_{233}	51	0	133
x_{112}	29	266	115	x_{234}	139	26	0
x_{113}	122	229	197	x_{241}	134	135	0
x_{114}	80	355	215	x_{242}	83	93	0
x_{121}	135	8	216	x_{243}	0	19	29
x_{122}	14	73	0	x_{244}	38	68	50
x_{123}	192	103	0	x_{251}	19	168	80
x_{124}	220	112	111	x_{252}	95	5	182
x_{131}	99	0	270	x_{253}	199	188	81
x_{132}	40	0	187	x_{254}	59	71	24
x_{133}	19	0	142	x_{311}	491	126	573
x_{134}	59	1	180	x_{312}	570	45	164
x_{141}	84	186	0	x_{313}	105	256	36
x_{142}	140	293	21	x_{314}	17	229	0
x_{143}	4	52	0	x_{321}	0	11	95
x_{144}	174	234	163	x_{322}	4	17	0
x_{151}	117	170	41	x_{323}	25	146	0
x_{152}	192	14	230	x_{324}	154	22	240
x_{153}	377	216	357	x_{331}	236	0	0
x_{154}	188	28	4	x_{332}	52	240	196
x_{211}	89	116	255	x_{333}	0	37	107
x_{212}	174	33	0	x_{334}	211	58	131
x_{213}	0	36	0	x_{341}	25	377	0
x_{214}	0	117	121	x_{342}	0	31	99
x_{221}	7	0	0	x_{343}	153	12	85
x_{222}	45	62	0	x_{344}	132	165	203
x_{223}	0	19	0	x_{351}	0	211	68
x_{224}	4	0	74	x_{352}	0	279	186
x_{231}	100	0	30	x_{353}	304	299	431
x_{232}	2	83	175	x_{354}	136	54	3

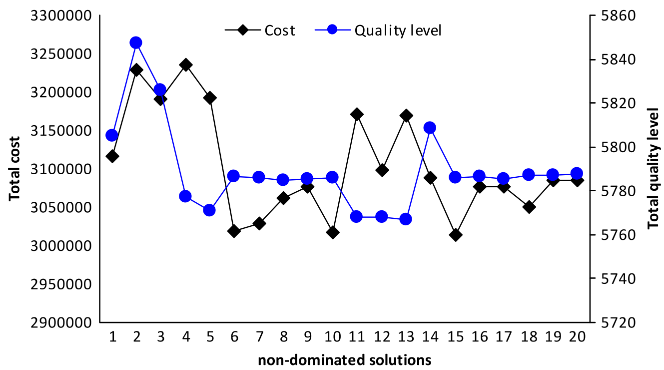

Fig. 2. Trade-off between total cost and total quality level objectives (model 1).

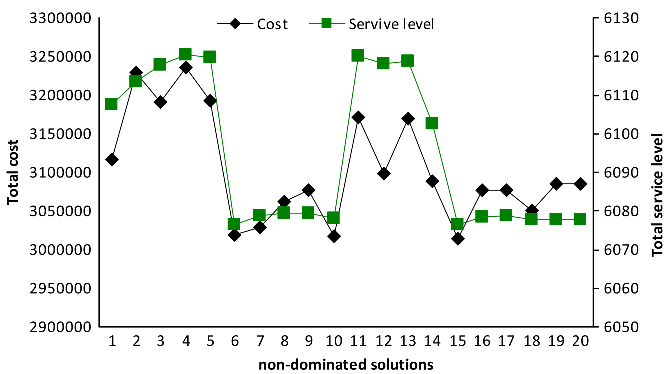

Fig. 3. Trade-off between total costs and total service level objectives (model 1).

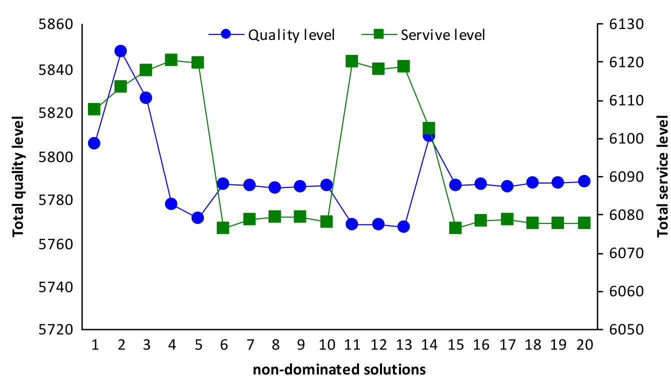

Fig. 4. Trade-off between total service level and total quality level objectives (model 1).

service level and total quality level, respectively. The figures indicate that a real trade-off exists between the total costs and total quality level (Fig. 2) and between the total cost and total service level (Fig. 3), as the lower values of total costs correspond to lower values of the total quality and service levels, and vice versa. For example, consider Sol.15, which is the best solution among 20 non-dominated solutions in terms of z_1, while at the same time being the solution with the lowest value in terms of total service level. In fact, the buyer should sacrifice total costs in favor of the other two objectives, and vice versa. This is also the case for total quality level and total service level (Fig. 4), as the higher values of total quality level corresponds to the lower values of total service level, and vice versa.

Table 6
Objective functions value of non-dominated solutions (model 2).

Solution no.	z_1	z_2	z_3
Sol.1	**2738 338**	**5818.396**	**6125.276**
Sol.2	2868 883	5872.331	6124.838
Sol.3	2784 008	5871.821	6124.622
Sol.4	2839 396	5872.181	6124.805
Sol.5	2929 908	5822.452	6133.526
Sol.6	2821 482	5872.301	6124.777
Sol.7	2901 355	5822.052	6133.544
Sol.8	2902 476	5872.601	6124.913
Sol.9	**2998 805**	**5837.062**	**6143.507**
Sol.10	2954 220	5835.483	6139.547
Sol.11	2937 151	5822.822	6133.396
Sol.12	2954 333	5835.635	6139.995
Sol.13	2954 209	5835.469	6139.506
Sol.14	2937 334	5873.001	6125.09
Sol.15	2987 624	5849.582	6136.774
Sol.16	2955 503	5844.621	6140.448
Sol.17	2954 323	5835.621	6139.955
Sol.18	2973 617	5826.92	6141.903
Sol.19	**2998 166**	**5873.361**	**6123.928**
Sol.20	2938 177	5828.932	6133.809

Table 7
Decision-making variables value of three non-dominated solutions (model 2).

x_{ijt}	Sol.1	Sol.9	Sol.19	x_{ijt}	Sol.1	Sol.9	Sol.19
x_{111}	281	225	0	x_{233}	0	167	241
x_{112}	0	0	2	x_{234}	0	0	194
x_{113}	29	241	0	x_{241}	0	0	0
x_{114}	550	380	550	x_{242}	0	0	0
x_{121}	0	0	0	x_{243}	0	0	0
x_{122}	0	0	0	x_{244}	234	0	0
x_{123}	0	0	0	x_{251}	0	0	0
x_{124}	0	386	0	x_{252}	0	0	175
x_{131}	0	0	0	x_{253}	157	381	0
x_{132}	0	0	0	x_{254}	0	142	109
x_{133}	0	0	0	x_{311}	95	0	179
x_{134}	0	307	420	x_{312}	0	331	114
x_{141}	0	0	0	x_{313}	580	0	346
x_{142}	238	27	0	x_{314}	637	578	63
x_{143}	890	773	0	x_{321}	0	0	0
x_{144}	362	0	0	x_{322}	0	0	0
x_{151}	0	0	0	x_{323}	0	0	245
x_{152}	101	117	0	x_{324}	0	409	36
x_{153}	0	0	767	x_{331}	0	0	36
x_{154}	0	15	755	x_{332}	292	0	0
x_{211}	0	0	527	x_{333}	0	0	575
x_{212}	0	3	0	x_{334}	0	30	207
x_{213}	536	309	0	x_{341}	0	0	49
x_{214}	0	246	0	x_{342}	0	0	0
x_{221}	0	0	0	x_{343}	0	44	242
x_{222}	0	0	0	x_{344}	24	185	0
x_{223}	0	0	0	x_{351}	1	0	0
x_{224}	0	4	0	x_{352}	294	393	0
x_{231}	0	0	0	x_{353}	691	0	397
x_{232}	296	0	0	x_{354}	0	659	120

4.2. Results (model 2)

In this paragraph, we discuss the results of model 2. Table 6 shows the objective functions value of a set of 20 non-dominated solutions and Table 7 displays three Pareto-optimal solutions. Although selecting a solution from the set of Pareto-optimal solutions depends on decision-maker's point of view, we select three solutions, for instance, in favor of each objective function, i.e. a solution with the lowest value of z_1 (Sol.1), a solution with the highest value of z_2 (Sol.19) and a solution with the highest value of z_3 (Sol.9).

Figs. 5–7 show the objective trade-off between total costs and total quality level, total costs and total service level and total

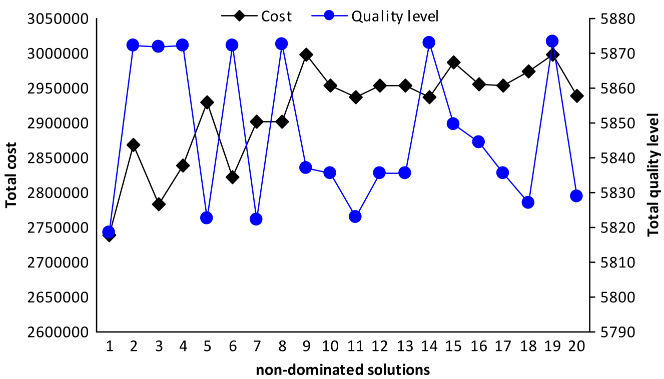

Fig. 5. Trade-off between total cost and total quality level objectives (model 2).

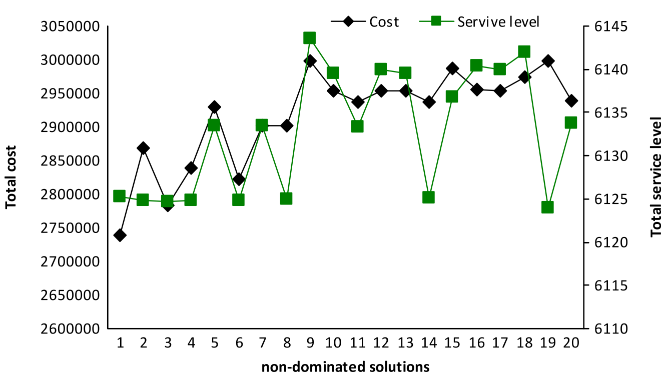

Fig. 6. Trade-off between total cost and total service level objectives (model 2).

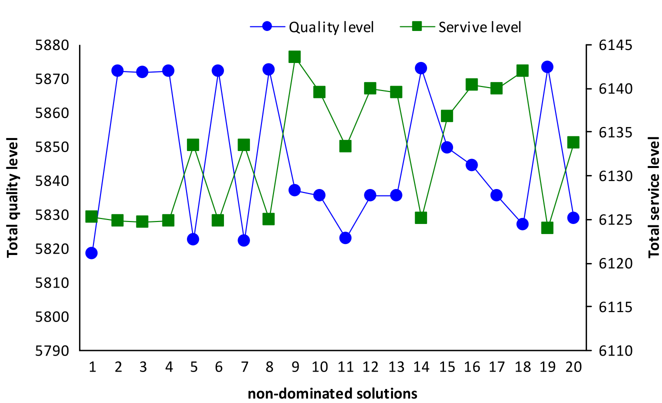

Fig. 7. Trade-off between total service level and total quality level objectives (model 2).

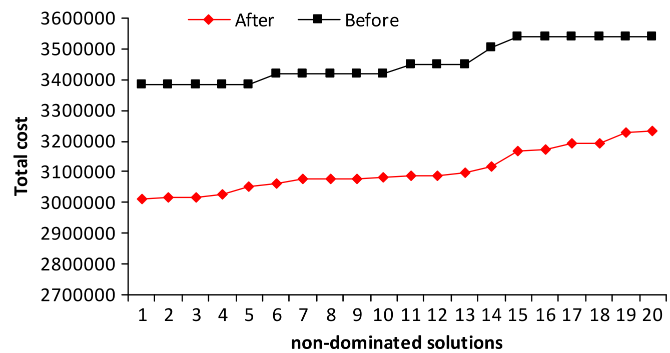

Fig. 8. Total costs; before and after considering the new formulation of ordering costs (model 1).

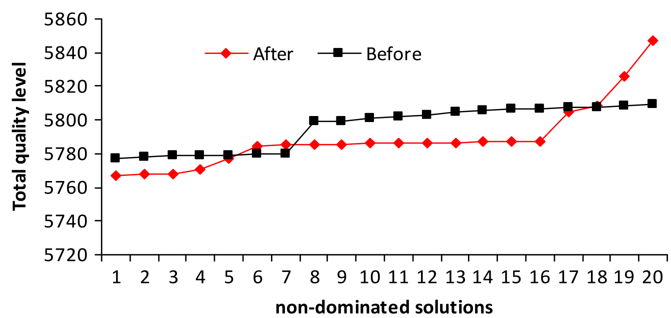

Fig. 9. Total quality level; before and after considering the new formulation of ordering costs (model 1).

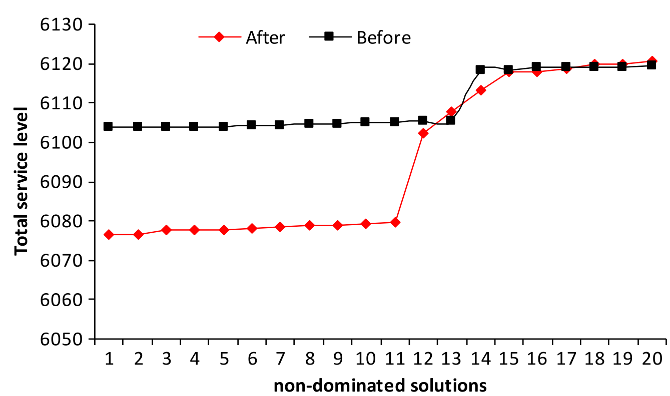

Fig. 10. Total service level; before and after considering the new formulation of ordering costs (model 1).

service level and total quality level, respectively. For this model, the trade-off between total costs and total service level and between total quality level and total service level is higher than the trade-off between total costs and total quality level.

4.3. Results comparison

In model 2, where backordering is allowed, the buyer can improve all three objectives, which means that the total costs of all non-dominated solutions of model 2 are lower than that of all non-dominated solutions of model 1. The highest total quality level in model 2 (5873.361, Sol.19) is higher than the highest total quality level in model 1 (5847.146, Sol.2), while the lowest total quality in model 2 (5818.396, Sol.1) is higher than the lowest total quality in model 1 (5767.074, Sol.13), and the total service level of all non-dominated solutions of model 2 are higher than that of all non-dominated solutions of model 1. In light of the fact that the backordering costs of all the products are higher than their holding costs, the superiority of model 2 has to do with the fact that the buyer has the option to buy more products in periods when prices are lower and/or the quality or service levels are higher. Although the same applies in model 1, there buyers cannot benefit from this opportunity in inverse direction as is possible in model 2. In other words, customers do not wait until prices are lower and/or quality improves. We also solved the problem with several different values for b_i and found that, as long as the backordering costs of *all* products are higher than their corresponding holding cost, model 2 shows a better performance with regard to objective functions. However, the superiority of model 2 is not maintained when the backordering costs of some or all products are lower than their corresponding holding costs.

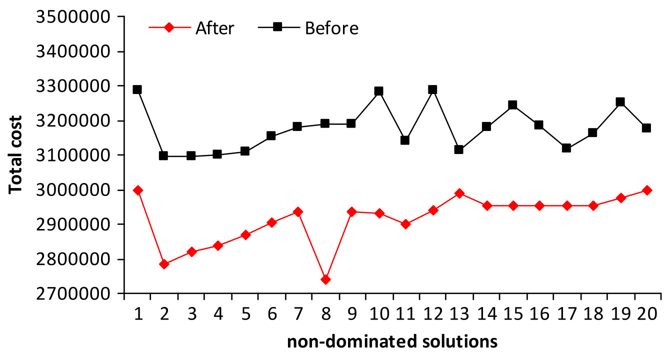

Fig. 11. Total costs; before and after considering the new formulation of ordering costs (model 2).

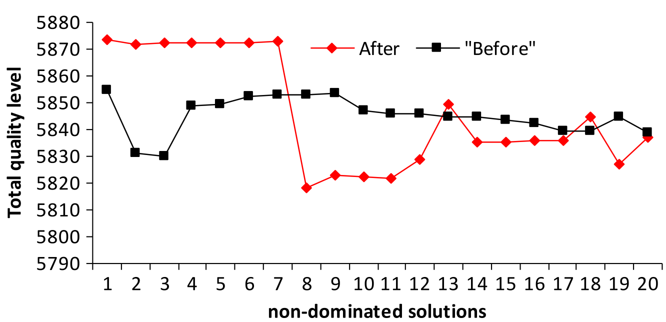

Fig. 12. Total quality level; before and after considering the new formulation of ordering costs (model 2).

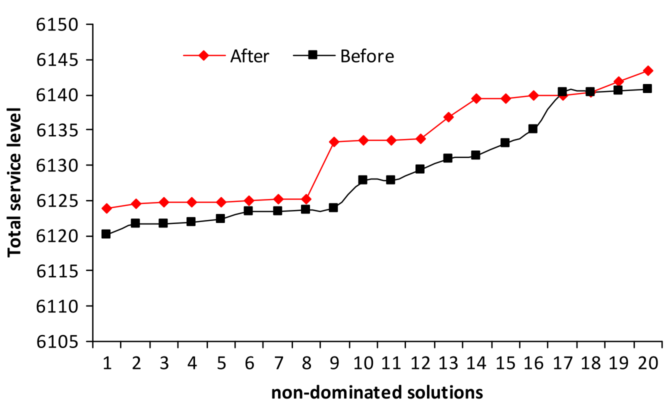

Fig. 13. Total service level; before and after considering the new formulation of ordering costs (model 2).

Here, we also look at the effect of the new formulation of ordering cost on the results compared to the traditional formulation. The second term of the first objective function (z_1) of both models is $\sum_j\sum_t o_j e^{-\beta_j \sum_{k=1}^{t} y_{jk}} y_{jt}$. As mentioned before, this formulation establishes an exponential relationship between order frequency ($\sum_{k=1}^{t} y_{jk}$) and ordering cost. β_j determines the intensity of this relationship, such that the higher value of β_j the greater the reduction in ordering costs. If $\beta_j=0$, the traditional formulation $\sum_j\sum_t o_j y_{jt}$ is obtained. For comparison purposes, we *set all* β_j to zero and compare two scenarios for both models.

Figs. 8–10 show this comparison for model 1. Note that we first sort the non-dominated solution in ascending order.

Figs. 11–13 show the comparison results in total costs, total quality level and total service level for 20 non-dominated solutions for model 2. Note that here we also begin by sorting these solutions in ascending order.

The results indicate that this change improves the value of the first objective function (total costs) of all non-dominated solutions in both models. However, this is not in support of the two other objective functions. Solving the models with several different data leads to the same conclusion. Because the new formulation mentioned earlier is part of the total cost function, it is wise not to see superiority or inferiority in the other two objective functions.

5. Conclusion and future work

Lot-sizing and supplier selection are two important decisions any buyer has to make. However, due to the inherent interdependency between these two decisions, the buyer cannot optimize them separately. To combine the two, the buyer needs to consider the main objectives with regard to supplier selection, such as cost minimization, and quality and service level maximization (Ghodsypour and O'Brien, 1998, 2001), and with regard to lot-sizing, things like as inventory costs minimization. To that end, in this paper, we have formulated two multi-objective optimization models based on two scenarios where shortage plays a role. The first model is geared towards situations where customers are not in the habit of wanting to wait and the buyer does not want to be faced with lost sales, while the second model is geared towards situations where customers accept late delivery. The results indicate that, if the customer agrees to late delivery (backordering) the minimum level of buyer's inventory costs is less than if the buyer has to meet all demands at the requested time.

In formulating the ordering costs and looking at the effect of order frequency on ordering costs, we have applied a new approach that creates a long term relationship between buyer and supplier. This does not mean that the buyer reduces the size of the orders he places with the suppliers to increase the order frequency and thereby reduce ordering costs. Instead, the buyer places orders with a limited number of suppliers more frequently, rather than dispersing the orders among a large number of suppliers.

In methodological terms, this paper introduces a novel, easy and flexible operator (the refiner operator) in the applied genetic algorithm dealing with the 'hard constraints' of the models. This operator not only speeds up the algorithm to find the feasible solutions but also improves the convergence of the solutions to the Pareto-regions while preserving population density.

This paper suggests a number of avenues for future research. In this paper, the supplier's service level and quality level are treated as time-dependent functions. Although this makes the model closer to real-world problems, as far as future research is concerned, we suggest examining the effects of other attributes of buyer–supplier relationships on supplier quality and service level, for instance information sharing, supply risk, power balance and joint investment in quality improvement. In addition, we suggest investigating problems associated with lost sales and the combination of lost sales and backordering conditions.

In this paper, the prices proposed by suppliers are assumed to be independent of the size of orders being placed. However, in some cases, suppliers are prone to give size-dependent discounts, which means that it may be interesting to investigate the problems discussed in this paper taking discounts into account, while at the same time taking a look at the relationship between price levels and customer demand.

References

Amid, A., Ghodsypour, S.H., O'Brien, C., 2006. Fuzzy multiobjective linear model for supplier selection in a supply chain. International Journal of Production Economics 104 (2), 394–407.

Anderson, E.T., Fitzsimons, G.J., Simester, D.I., 2006. Measuring and mitigating the cost of stockouts. Management Science 52 (11), 1751–1763.

Augusto, O.B., Rabeau, S., Depince, Ph., Bennis, F., 2006. Multi-objective genetic algorithms: a way to improve the convergence rate. Engineering Applications of Artificial Intelligence 19 (5), 501–510.
Basnet, C., Leung, J.M.Y., 2005. Inventory lot-sizing with supplier selection. Computers and Operations Research 32 (1), 1–14.
Ben-Daya, M., Darwish, M., Ertogral, K., 2008. The joint economic lot sizing problem: review and extensions. European Journal of Operational Research 185 (2), 726–742.
Berden, T.P.J., Brombacher, A.C., Sander, P.C., 2000. The building bricks of product quality: an overview of some basic concepts and principles. International Journal of Production Economics 67 (1), 3–15.
Bitran, G.R., Yanasse, H.H., 1982. Computational complexity of the capacitated lot size problem. Management Science 28 (10), 1174–1186.
Centikaya, S., Parlar, M., 1998. Non linear programming analysis to estimate implicit inventory backorder costs. Journal of Optimization Theory and Applications 97 (1), 71–92.
Dai, T., Qi, X., 2007. An acquisition policy for a multi-supplier system with a finite-time horizon. Computers & Operations Research 34 (9), 2758–2773.
Deb, K., 2001. Multi-Objective Optimization using Evolutionary AlgorithmsWiley, Chichester, UK.
Deb, K., Agrawal, S., Pratap, A., Meyarivan, T., 2000. A fast elitist non-dominated sorting genetic algorithm for multi-objective: NSGA-II. In: Proceedings of the Parallel Problem Solving from Nature VI Conference, Springer-Verlag, pp. 846–858.
Dellaert, N., Jeunet, J., Jonard, N., 2000. A genetic algorithm to solve the general multi-level lot-sizing problem with time-varying costs. International Journal of Production Economics 68 (3), 241–257.
Disney, S.M., Naim, M.M., Towill, D.R., 2000. Genetic algorithm optimisation of a class of inventory control systems. International Journal of Production Economics 68 (3), 259–278.
Ertogral, K., Darwish, M., Ben-Daya, M., 2007. Production and shipment lot sizing in a vendor–buyer supply chain with transportation cost. European Journal of Operational Research 176 (3), 1592–1606.
Florian, M., Lenstra, J.K., Rinnooy Kan, A.H.G., 1980. Deterministic production planning: algorithms and complexity. Management Science 26 (7), 669–679.
Ghodsypour, S.H., O'Brien, C., 1998. A decision support system for supplier selection using an integrated analytic hierarchy process and linear programming. International Journal of Production Economics 56–57, 199–212.
Ghodsypour, S.H., O'Brien, C., 2001. The total cost of logistics in supplier selection, under conditions of multiple sourcing, multiple criteria and capacity constraint. International Journal of Production Economics 73 (1), 15–27.
Goldberg, D.E., 1989. Genetic Algorithms in Search, Optimization, and Machine Learning. Addison Wesley.
Gollner, P., Dell and HP see laptop battery shortage, 26 March 2008, 〈http://www.reuters.com/article/idUSN2540920320080326〉 (retrieved on 01-09-2010).
Gupta, R.K., Bhunia, A.K., Goyal, S.K., 2007. An application of genetic algorithm in a marketing oriented inventory model with interval valued inventory costs and three-component demand rate dependent on displayed stock level. Applied Mathematics and Computation 192 (2), 466–478.
Homaifar, A., Lai, S.H.-V., Qi, X., 1994. Constrained optimization via genetic algorithms. Simulation 62 (4), 242–254.
Jiang, Z., Xuanyuan, S., Li, L., Li, Z., 2010. Inventory-shortage driven optimisation for product configuration variation. International Journal of Production Research, first published on: 05 March 2010 (iFirst), doi:10.1080/00207540903555494.
Jimenez, F., Verdegay, J.L., Gomez-Skarmeta, A.F., 1999. Evolutionary techniques for constrained multiobjective optimization problems. In: Proceedings of the Workshop on Multi-Criterion Optimization Using Evolutionary Methods held at Genetic and Evolutionary Computation Conference (GECCO-1999), pp. 115–116.
Karimi, B., Fatemi Ghomi, S.M.T., Wilson, J.M., 2003. The capacitated lot sizing problem: a review of models and algorithms. Omega 31 (5), 365–378.
Lambert, D.M. (Ed.), 3rd edition Supply Chain Management Institute, Sarasota, Florida.
Liao, Z., Rittscher, J., 2007. A multi-objective supplier selection model under stochastic demand conditions. International Journal of Production Economics 105 (1), 150–159.
Liberopoulos, G., Tsikis, I., Delikouras, S., 2010. Backorder penalty cost coefficient "b": What could it be? International Journal of Production Economics 123 (1), 166–178.
Mentzer, J.T., DeWitt, W., Keebler, J.S., Min, S., Nix, N.W., Smith, C.D., Zacharia, Z.G., 2001. Defining supply chain management. Journal of Business Logistics 22 (2), 1–25.
Meyr, H., Wagner, M., Rohde, J., 2008. Structure of advanced planning systems. In: Stadtler, H., Kilger, C. (Eds.), Supply Chain Management and Advanced Planning—Concepts, Models, Software, and Case Studies. Springer, Berlin, pp. 109–115.
Michalewicz, Z., 1994. Genetic Algorithms+Data Structures=Evolution Programs. AI Series. Springer Verlag, New York.
Michalewicz, Z., Schoenauer, M., 1996. Evolutionary algorithms for constrained parameter optimization problems. Evolutionary Computation Journal 4 (1), 1–32.
Morgan, R., Hunt, S., 1994. The commitment–trust theory of relationship marketing. Journal of Marketing 58, 20–38.
Rezaei, J., Davoodi, M., 2005. Multi-item Fuzzy Inventory Model with Three Constraints: Genetic Algorithm Approach. Lecture Notes in Artificial Intelligence, vol. 3809. Springer-Verlag, Berlin, Heidelberg, pp. 1120–1125.
Rezaei, J., Davoodi, M., 2006. Genetic Algorithm for Inventory Lot-sizing with Supplier Selection under Fuzzy Demand and Costs. Advances in Applied Artificial Intelligence, vol. 4031. Springer-Verlag, Berlin, Heidelberg, pp. 1100–1110.
Rezaei, J., Davoodi, M., 2008. A deterministic, multi-item inventory model with supplier selection and imperfect quality. Applied Mathematical Modelling 32 (10), 2106–2116.
Robinson, P., Narayanan, A., Sahin, F., 2009. Coordinated deterministic dynamic demand lot-sizing problem: a review of models and algorithms. Omega 37 (1), 3–15.
Sadeghi Moghadam, M.R., Afsar, A., Sohrabi, B., 2008. Inventory lot-sizing with supplier selection using hybrid intelligent algorithm. Applied Soft Computing 8 (4), 1523–1529.
Schneider, H., 1981. Effect of service-levels on order-points or order-levels in inventory models. International Journal of Production Research 19 (6), 615–631.
Sharafali, M., Co, H.C., 2000. Some models for understanding the cooperation between the supplier and the buyer. International Journal of Production Research 38 (15), 3425–3449.
Spekman, R.E., Kamauff Jr, J.W., Myhr, N., 1998. An empirical investigation into supply chain management: a perspective on partnerships. International Journal of Physical Distribution & Logistics Management 28 (8), 630–650.
Srinivas, N., Deb, K., 1994. Multi-objective optimization using non-dominated sorting in genetic algorithms. Journal of Evolutionary Computation 2 (3), 221–248.
van Hop, N., Tabucanon, M.T., 2005. Adaptive genetic algorithm for lot-sizing problem with self-adjustment operation rate. International Journal of Production Economics 98 (2), 129–135.
van Norden, L., van de Velde, S., 2005. Multi-product lot-sizing with a transportation capacity reservation contract. European Journal of Operational Research 165 (1), 127–138.
Wagner, H.M., 2004. Comments on "Dynamic Version of the Economic Lot Size Model". Management Science 50 (12), 1775–1777.
Wagner, H.M., Whitin, T.M., 1958. Dynamic version of the economic lot size model. Management Science 5 (1), 89–96.
Walsh, L., Dell hit by server inventory shortages, Channelinsider, 14-01-2010, 〈http://www.channelinsider.com/c/a/Dell/Dell-Hit-by-Server-Inventory-Shortages-100399/〉 (retrieved on 01-09-2010).
Woo, Y.Y., Hsu, S.L., Wu, S., 2001. An integrated inventory model for a single vendor and multiple buyers with ordering cost reduction. International Journal of Production Economics 73 (3), 203–215.
Xie, J., Dong, J., 2002. Heuristic genetic algorithms for general capacitated lot-sizing problems. Computers & Mathematics with Applications 44 (1–2), 263–276.
Xu, H., 2010. Managing production and procurement through option contracts in supply chains with random yield. International Journal of Production Economics 126 (2), 306–313.
Zitzler, E., Laumanns, M., Thiele, L., 2001. SPEA2: Improving the strength Pareto evolutionary algorithm for multi-objective optimization. Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems, 95–100.