ORIGINAL PAPER

A supply chain design problem with facility location and bi-objective transportation choices

Elias Olivares-Benitez ·
José Luis González-Velarde ·
Roger Z. Ríos-Mercado

Abstract A supply chain design problem based on a two-echelon single-product system is addressed. The product is distributed from plants to distribution centers and then to customers. There are several transportation channels available for each pair of facilities between echelons. These transportation channels introduce a cost–time tradeoff in the problem that allows us to formulate it as a bi-objective mixed-integer program. The decisions to be taken are the location of the distribution centers, the selection of the transportation channels, and the flow between facilities. Three variations of the classic ε-constraint method for generating optimal Pareto fronts are studied in this paper. The procedures are tested over six different classes of instance sets. The three sets of smallest size were solved completely obtaining their efficient solution set. It was observed that one of the three proposed algorithms consistently outperformed the other two in terms of their execution time. Additionally, four schemes for obtaining lower bound sets are studied. These schemes are based on linear programming relaxations of the model. The contribution of this work is the introduction of a new bi-objective optimization problem, and a computational study of the ε-constraint methods for obtaining optimal efficient fronts and the lower bounding schemes.

E. Olivares-Benitez
UPAEP University, 21 Sur 1103, Puebla, Puebla 72410, Mexico
e-mail: Elias.Olivares@upaep.mx

J.L. González-Velarde
Tecnologico de Monterrey, E. Garza Sada Sur 2501, Monterrey, NL 64849, Mexico
e-mail: Gonzalez.Velarde@itesm.mx

R.Z. Ríos-Mercado (✉)
Graduate Program in Systems Engineering, Universidad Autónoma de Nuevo León, AP 111-F,
Cd. Universitaria, San Nicolás de los Garza, NL 66450, Mexico
e-mail: roger@mail.uanl.mx

Mathematics Subject Classification (2000) 90-08 · 90B06 · 90B10 · 90B80 · 90C11 · 90C29

1 Introduction

Supply chains have received much attention recently after recognizing the importance of the logistics costs in the cost structure of the products. Their efficient management provides a competitive advantage to domestic and international firms. At the strategic level, managers must design the supply chain to achieve the minimum cost and to meet a desired customer service level. This task concerns aspects of inventory, transportation and facility location (Ballou 1999). The supply chain is composed of facilities and transportation flows between facilities. These facilities perform different roles as suppliers, plants, warehouses, distribution centers and retailers. Thus, the decisions implied in supply chain design are (Simchi-Levi et al. 2000):

- To determine the number of facilities
- To determine the location of the facilities
- To determine the capacities of the facilities
- To allocate products to facilities
- To determine the flow of products between facilities

Network design decisions determine the supply chain configuration and have a significant impact in logistics costs and responsiveness (Chopra and Meindl 2004). For instance, facility location has a long term impact in the supply chain because of the high cost to open a facility or to move it. Opening and inventory costs induce to reduce the number of facilities, while responsiveness causes a contrary effect. Responsiveness is the delivery of products in a short lead time, achieved by having short transportation time, availability of the product in inventory, or a combination of both. A high number of facilities may reduce the time to deliver a product to the final customer. In certain products, lead time can be viewed as an added value so that the firm that makes them available first can obtain short and long term advantages in the market.

Many models developed to design distribution systems are based on discrete facility location where a set of potential sites is known. The earliest models were formulated by Baumol and Wolfe (1958), and Kuehn and Hamburger (1963). These and subsequent models have been formulated as mixed-integer programming problems. The evolution of such models has considered until recently (Klose and Drexl 2005) the following elements:

- Number of echelons
- Facility capacity
- Number of products
- Time periods
- Stochastic demand
- Side constraints to include:
  - Single or multiple sourcing
  - Routing

However, an area of opportunity in supply chain design is to address the influence of the transportation channels. The decision to use a certain transportation channel has an effect on the lead time to deliver a product which is often an indicator of customer service level. The availability of different channels to transport the product between a pair of facilities is a feature of modern logistics services. These transportation channels can be seen as transportation modes (rail, truck, ship, airplane, etc.), shipping services (express, normal, overnight, etc.) or just as simple as the offer from different companies. Transportation choices are differentiated by parameters of time and cost. Commonly, these parameters are negatively correlated with shorter times for the most expensive alternatives. For many years, distance was treated as surrogate of transportation cost and time. Nowadays this is not a valid assumption.

In this paper, we introduce a problem for supply chain design of a two-echelon distribution system. We include the decision of selecting the transportation channel between each pair of facilities. The problem is treated as a bi-objective optimization problem where cost and time criteria are minimized. This problem has been named "Capacitated Fixed Cost Facility Location Problem with Transportation Choices" (CFCLP-TC). A literature review is presented in Sect. 2. Section 3 shows the problem description in detail. The mathematical model is explained in Sect. 4. Section 5 is dedicated to presenting three algorithms to obtain the set of efficient solutions for an instance. These algorithms are variations of the ε-constraint method. The algorithms are compared later in terms of efficiency (run time) and solution quality. Four schemes for obtaining lower bound sets are described in Sect. 6. These schemes are based on linear relaxations of the model and on the information of the mixed-integer program (MIP) solutions. Therefore, to define their quality they are used for small instances where the set of efficient solutions is known. Section 7 shows the results of the computational experience with the algorithms proposed and the lower bound schemes studied. The final conclusions are presented in Sect. 8.

2 Literature review

Several reviews about models for supply chain design (Aikens 1985; Thomas and Griffin 1996; Vidal and Goetschalckx 1997; Klose and Drexl 2005; Sahin and Sural 2007; Melo et al. 2009) exist in the literature. In general, the optimization models described were formulated as mixed-integer programs and solved by decomposition techniques and heuristic methods. However, as mentioned before, the influence of the availability of different transportation channels between facilities has not been studied in depth. The transportation choices are qualified in terms of time and cost producing a tradeoff that affects the distribution network configuration. This feature induces naturally to re-formulate the supply chain design problem as a bi-objective optimization model. Looking at the review by Current et al. (1990), it is evident that the balance of these measures has not been studied extensively. Just until recently, Farahani et al. (2009) dedicate a review to the multicriteria models related to facility location problems. Some relevant works produced recently are highlighted next.

2.1 Cost–time tradeoff with one transportation mode between facilities

A first set of works includes those that identify the cost–time tradeoff as an important element in supply chain design. However, these papers do not relate this balance to the availability of transportation choices between facilities. Zhou et al. (2003) introduced two separate objective functions for cost and time. The total sum of transportation times is minimized in one objective function. In that paper, only one transportation alternative between each pair of warehouse–customer nodes is considered. The facility location decision is not included in the model. A genetic algorithm is used to construct a set of non-dominated solutions. Eskigun et al. (2005) use an aggregated function for time and cost. Although different transportation modes are included in their model (rail and truck), the problem is to select between a direct and an intermodal shipping strategy. They do not have transportation choices between each pair of locations. A Lagrangian heuristic is developed to solve the mixed-integer programming model. In the problem proposed by Truong and Azadivar (2005), the cost–time tradeoff is recognized. In this case, the time measure is included in the objective function as a parameter that influences inventory cost. Lead time is based on the complete production–distribution path for the product without transportation alternatives between nodes. Their solution approach is based on an iterative method that uses a hybrid genetic algorithm. In an intermediate stage of the algorithm, a mixed-integer programming model and a simulation model are created and solved. The results are used as entries for the external cycle of the genetic algorithm. Altiparmak et al. (2006) handle transportation time as a constraint. In their problem, a set of feasible distribution centers is selected a priori. These facilities are those that are able to deliver the product to the customer before a time limit. The selection of a transportation mode is not considered in their model. Three objective functions are proposed to minimize total cost, to maximize total customer demand satisfied, and to minimize the unused capacity of distribution centers. In this case, a genetic algorithm is used to obtain a set of non-dominated solutions. Vidyarthi et al. (2009) analyze a model based on a response time of distribution centers that serve a set of customers. The response time is modeled using a queuing formulation and is not tied to the distance between distribution centers and customers. The authors developed a Lagrangian heuristic to solve the problem.

2.2 Multiple transportation modes between facilities without cost–time tradeoff

A second set of papers appreciates the influence of the selection of the transportation channel in the distribution network design. The main feature of this set of works is that the cost–time tradeoff is not related to that decision. In the model proposed by Benjamin (1990), the selection of the transportation mode is based on the capacity of the channel. The time parameter is not considered. The problem is formulated as a non-linear programming model that minimizes an aggregated cost objective function. This function combines transportation and inventory costs. The solution procedure is a heuristic method based on the Benders decomposition technique. Wilhelm et al. (2005) also include the selection of a transportation mode but it is based on cost. In this case, a portion of the quantity transported can be assigned to different transportation modes. It is a multi-period problem where the supply chain configuration

changes dynamically. A set of scenarios were solved to optimality for relatively small instances. Cordeau et al. (2006) presented a model that has many of the elements of classic distribution network design. Additionally, the selection of a transportation mode is one of the decisions in the model. Nevertheless, the cost–time tradeoff is not studied. The selection of the transportation mode is based on capacity and cost. They developed a Benders' decomposition method, which is a technique commonly used to solve such type of problems.

2.3 Cost–time tradeoff with multiple transportation modes between facilities

The last set of works recognizes the importance of the cost–time tradeoff and relates it to the availability of transportation choices between facilities. In the paper by Arntzen et al. (1995), the cost–time balance is handled as a weighted combination in the objective function. The decision is made on the quantity to be sent through each transportation mode available. Here, transportation time is a linear function of the quantity shipped. The problem is solved using elastic penalties for violating constraints, and a row-factorization technique. In the work by Zeng (1998), the importance of the lead time–cost tradeoff is emphasized. This feature is directly associated to the transportation modes available between pairs of nodes in the network. The author proposed a mathematical model to optimize both measures in the process of supply chain design. A main difference with traditional models is that facility location is not addressed. A dynamic programming algorithm was presented to construct the efficient frontier assuming the discretization of time measure. Graves and Willems (2005) propose a model that aggregates cost and time in the objective function. Their approach is not directly related to transportation modes but it is open to any alternative to be chosen at each stage of the network design process. They use a dynamic programming algorithm to solve this problem. Chan et al. (2006) present a multiobjective model that optimizes a combined objective function with weights. Some of the criteria include cost and time functions. In this case, the objective function for time is composed of many sources in addition to transportation time. The selection of the transportation channel is associated to the cost–time tradeoff. It is a complex model that includes stochastic components. However, in this case facility location is not considered. Similarly to other approaches, the transportation time is a linear function of the quantity transported. In this case, a genetic algorithm is the base of an iterative method to solve the problem for several weight scenarios. ElMaraghy and Majety (2008) propose a model that optimizes cost, including the cost of late delivery. The model is formulated in terms of periods to include the dynamic nature of the decisions. The authors use commercial optimization software to solve the model, and analyze different scenarios.

2.4 Remarks

It is evident that there are few works that handle the cost–time tradeoff derived from the transportation channel selection in the supply chain design problem. The scarce models proposed in the literature are different from the CFCLP-TC model presented. Those models make some assumptions that facilitate the solution of the problem, but

that are unreal or inconvenient for the decision making process. The first assumption is the linearization of time. While this assumption helps to make the problem easier to solve, it is not adequate to represent real transportation conditions. Usually, the time to transport a product is independent of the quantity to be shipped. In the negotiation of a transportation service, the quantity may affect cost because of economy of scale but time is not affected. Hence the transportation time is tied directly to the transportation channel and it is independent of the quantity transported. The second assumption is the preference for some criterion, usually the cost objective. The use of this assumption is implicit when the multiobjective problem is transformed to a single-objective problem combining the cost and time criteria in an aggregated (utility) function. It helps also to reduce the complexity of the problem, but it is inconvenient for the decision making process because some times the preferences of the decision maker are not known a priori. Additionally, the selection of the appropriate weights is a difficult task for criteria with different measures like cost and time that cannot be compared directly. Hence an approach to show different non-dominated solutions as alternatives to the decision maker is eluded, while it may be a good choice when his/her preference is not known or the criteria cannot be compared easily. These assumptions were excluded from the formulation and the solution approach to obtain a more realistic model capable of providing a set of alternatives to the decision maker.
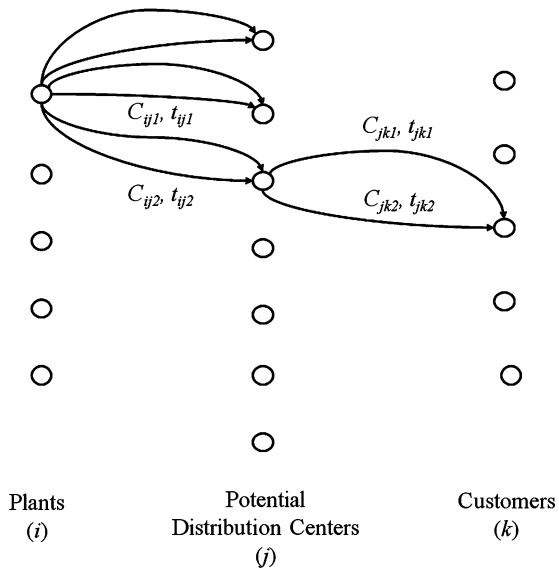
3 Problem description

The "Capacitated Fixed Cost Facility Location Problem with Transportation Choices" (CFCLP-TC) is based on a two-echelon system for the distribution of one product in a single time period. In the first echelon, the manufacturing plants send product to distribution centers. The second echelon corresponds to the flow of product from the distribution centers to the customers. The number and location of plants and customers are known.

There is a set of potential locations to open distribution centers. The number of open distribution centers is not defined a priori. Each candidate site has a fixed cost for opening a facility. Each potential site has also a limited capacity. This capacity is related to dispatching rate which depends on factors like physical limits, equipment and productivity of the facility. The plants have limited manufacturing capacity. This capacity represents the production rate at each plant. A supply constraint, called single source constraint, states that each customer is supplied by a single distribution center. However, the demand of each customer must be met.

The most important feature added to the problem, and one of the main contributions of this work is to consider several alternatives to transport the product from one facility to the other in each echelon of the network. Each option represents a type of service with associated cost and time parameters. The existence of third party logistics companies (3PL) makes available different transportation services in the market. The alternatives are generated by the offer from different companies, the availability of different types of service at each company (e.g., express and regular), and the use of different modes of transportation (e.g., truck, rail, airplane, ship or inter-modal). Commonly, these differences involve an inverse correspondence between time and

Fig. 1 Single product, single period, and two-echelon distribution system. Each transportation channel has an associated time and unitary cost

C_{ij1}, t_{ij1}

C_{jk1}, t_{jk1}

C_{ij2}, t_{ij2}

C_{jk2}, t_{jk2}

Plants
(i)

Potential
Distribution Centers
(j)

Customers
(k)

cost, i.e., a faster service will be more expensive. A scheme of the distribution network is shown in Fig. 1.

The idea of this problem is to select the appropriate sites to open distribution centers and the flow between facilities to minimize the combined cost of transportation and facility opening. This problem is very common in distribution networks where manufacturing plants and points of demand already exist. However, this problem also includes the selection of the transportation channel. This decision has an impact on the transportation time from the plant to the customer. The tradeoff between cost and time must be considered in the formulation of a mathematical model that minimizes both criteria simultaneously. Hence, the problem should be addressed with a bi-objective optimization model. Following this approach, one criterion minimizes the combined cost of transportation and facility location. The other criterion looks for the minimum time to transport the product along the path from the plant to the customer.

Some assumptions have been made and some elements have been left out of the problem to maintain the tractability. Inventory costs, production costs, capacity on the transportation links, congestion times at the facilities, international supply chain aspects, and non-linear transportation costs are not considered. Therefore, we have assumed that distribution centers do not retain inventory and their function is only to split the product received from the plants. As no inventory is retained at the distribution centers, the latter may be used as cross-docks for the case of multiple products. The distribution centers may receive product from any plant. There is no incentive to ship the product directly from the plants to the customers. Also, transshipment between facilities at the same stage is not allowed. Locally, at each pair of origin–destination nodes the transportation channels with dominated parameters are eliminated, i.e., those with coincident longer time and greater cost than any other. The sum of the capacity of the plants is enough to satisfy the total demand. The sum of

the capacity of the potential distribution centers is enough to satisfy the total demand. An important assumption is the discretization of the time parameter. Therefore, given reasonable times for transportation (hours, days) the values can be represented as integer units.

4 Mathematical framework

4.1 Model and notation

The CFCLP-TC problem described previously is represented as a bi-objective mixed-integer programming model. The model formulation is preceded by the notation shown below.

Sets:

I: set of plants i
J: set of potential distribution centers j
K: set of customers k
LP_{ij}:   set of arcs l between nodes i and j; $i \in I$, $j \in J$
LW_{jk}: set of arcs l between nodes j and k; $j \in J$, $k \in K$

Parameters:

CP_{ijl}:   cost of transporting one unit of product from plant i to distribution center j using arc ijl; $i \in I$, $j \in J$, $l \in LP_{ij}$
CW_{jkl}: cost of sending one unit of product from distribution center j to customer k using arc jkl; $j \in J$, $k \in K$, $l \in LW_{jk}$
TP_{ijl}:   time for transporting any quantity of product from plant i to distribution center j using arc ijl; $i \in I$, $j \in J$, $l \in LP_{ij}$
TW_{jkl}: time for transporting any quantity of product from distribution center j to customer k using arc jkl; $j \in J$, $k \in K$, $l \in LW_{jk}$
MP_i: capacity of plant i; $i \in I$
MW_j: capacity of distribution center j; $j \in J$
D_k: demand of customer k; $k \in K$
F_j: fixed cost for opening distribution center j; $j \in J$

Decision variables:

X_{ijl}:   quantity transported from plant i to distribution center j using arc ijl; $i \in I$, $j \in J$, $l \in LP_{ij}$
Y_{jkl}:   quantity transported from distribution center j to customer k using arc jkl; $j \in J$, $k \in K$, $l \in LW_{jk}$
Z_j: binary variable equal to 1 if distribution center j is open and equal to 0 otherwise; $j \in J$
A_{ijl}:   binary variable equal to 1 if arc ijl is used to transport product from plant i to distribution center j and equal to 0 otherwise; $i \in I$, $j \in J$, $l \in LP_{ij}$
B_{jkl}:   binary variable equal to 1 if arc jkl is used to transport product from distribution center j to customer k and equal to 0 otherwise; $j \in J$, $k \in K$, $l \in LW_{jk}$

Auxiliary variables:

T: maximum time that takes sending product from any plant to any customer

E_j^1: maximum time in the first echelon of the supply chain for active distribution center j, i.e., $E_j^1 = \max_{i,l}(TP_{ijl}A_{ijl}); i \in I, j \in J, l \in LP_{ij}$

E_j^2: maximum time in the second echelon of the supply chain for active distribution center j, i.e., $E_j^2 = \max_{k,l}(TW_{jkl}B_{jkl}); j \in J, k \in K, l \in LW_{jk}$

MODEL 1:

$$\min(f_1, f_2),$$

$$f_1 = \sum_{i \in I} \sum_{j \in J} \sum_{l \in LP_{ij}} CP_{ijl} X_{ijl} + \sum_{j \in J} \sum_{k \in K} \sum_{l \in LW_{jk}} CW_{jkl} Y_{jkl} + \sum_{j \in J} F_j Z_j, \quad (1)$$

$$f_2 = T \quad (2)$$

subject to

$$T - E_j^1 - E_j^2 \geq 0, \quad j \in J, \quad (3)$$

$$E_j^1 - TP_{ijl} A_{ijl} \geq 0, \quad i \in I, j \in J, l \in LP_{ij}, \quad (4)$$

$$E_j^2 - TW_{jkl} B_{jkl} \geq 0, \quad j \in J, k \in K, l \in LW_{jk}, \quad (5)$$

$$\sum_{j \in J} \sum_{l \in LW_{jk}} Y_{jkl} = D_k, \quad k \in K, \quad (6)$$

$$\sum_{j \in J} \sum_{l \in LP_{ij}} X_{ijl} \leq MP_i, \quad i \in I, \quad (7)$$

$$MW_j Z_j - \sum_{k \in K} \sum_{l \in LW_{jk}} Y_{jkl} \geq 0, \quad j \in J, \quad (8)$$

$$\sum_{i \in I} \sum_{l \in LP_{ij}} X_{ijl} - \sum_{k \in K} \sum_{l \in LW_{jk}} Y_{jkl} = 0, \quad j \in J, \quad (9)$$

$$\sum_{j \in J} \sum_{l \in LW_{jk}} B_{jkl} = 1, \quad k \in K, \quad (10)$$

$$\sum_{l \in LP_{ij}} A_{ijl} \leq 1, \quad i \in I, j \in J, \quad (11)$$

$$\sum_{l \in LW_{jk}} B_{jkl} \leq 1, \quad j \in J, k \in K, \quad (12)$$

$$X_{ijl} - A_{ijl} \geq 0, \quad i \in I, j \in J, l \in LP_{ij}, \quad (13)$$

$$Y_{jkl} - B_{jkl} \geq 0, \quad j \in J, k \in K, l \in LW_{jk}, \quad (14)$$

$$MP_i A_{ijl} - X_{ijl} \geq 0, \quad i \in I, j \in J, l \in LP_{ij}, \quad (15)$$

$$MW_j B_{jkl} - Y_{jkl} \geq 0, \quad j \in J, k \in K, l \in LW_{jk}, \quad (16)$$

$$\sum_{i \in I} \sum_{l \in LP_{ij}} A_{ijl} - Z_j \geq 0, \quad j \in J, \tag{17}$$

$$T, E_j^1, E_j^2, X_{ijl}, Y_{jkl} \geq 0, \quad i \in I, j \in J, k \in K, l \in LP_{ij}, l \in LW_{jk}, \tag{18}$$

$$Z_j, A_{ijl}, B_{jkl} \in \{0, 1\}, \quad i \in I, j \in J, k \in K, l \in LP_{ij}, l \in LW_{jk}. \tag{19}$$

In this formulation, objective function (1) minimizes the sum of the transportation cost and the cost for opening distribution centers. Objective function (2) minimizes the maximum time required to transport the product from the plants to the customers through each distribution center. This function was reformulated from (20) to eliminate the non-linearity:

$$f_2 = \max_j \left(\max_{i,l} (TP_{ijl} A_{ijl}) + \max_{k,l} (TW_{jkl} B_{jkl}) \right). \tag{20}$$

Constraints (3)–(5) complete the linearization of (20) into objective function (2). It is evident that variables A_{ijl} and B_{jkl} should take the value of 1 only for open distribution centers. Therefore, (20) considers the paths through open distribution centers and minimizes the maximum time along the supply chain. Constraints (6) require the demand satisfaction of each customer. Constraints (7) are formulated for not exceeding the capacity limits of the plants. Constraints (8) state that the flow going out from a distribution center must not exceed the capacity of the facility, but at the same time requires that the flow of product only can be done through open distribution centers. Constraints (9) keep the flow balance at the distribution center. Constraints (10) establish that each customer must be supplied by a single source. At most one arc may be selected between nodes i–j and nodes j–k, as required in constraints (11) and (12), respectively. Constraints (13)–(17) are formulated to make an appropriate link between the sets of variables A_{ijl}, B_{jkl}, X_{ijl}, Y_{jkl} and Z_j. When solving for f_1 as main objective function, the model tries to minimize the values of X_{ijl}, Y_{jkl} and Z_j. In the other direction, when solving for f_2 the model tries to minimize the values of A_{ijl}, B_{jkl}. Constraints (13) and (14) require that an arc must be inactive if it does not have flow through it. This avoids overestimating A_{ijl} and B_{jkl} when solving for f_1. The flow of product only can be done through active arcs as stated in constraints (15) and (16). Constraints (17) ensure that a distribution center must be closed if it has no active incident arcs on it, i.e., the distribution center does not receive flow of product. In this way, when solving for f_2 the model does not overestimate Z_j and avoids passing flow through inactive arcs. Constraints (18) are for continuous non-negative variables. Binary variables are required in constraints (19). If demands and capacities have integer values, it is not necessary to change constraint (18) to require integer values for X_{ijl} and Y_{jkl}. This is because once the values for A_{ijl}, B_{jkl} and Z_j are fixed, the remaining structure is a transportation problem. It is well known that the unimodularity property of this problem produces these integral results under such a condition.

It should be noted that constraints (10) imply that a problem has a feasible solution only if for each customer there exists at least one distribution center with enough capacity to satisfy its demand. Also note that constraints (10) imply (12), and constraints (13)–(14) and (17) are redundant in a strict sense. However, these strengthen

the model when solving the subproblems or relaxations so they are included in the model.

The CFCLP-TC is NP-hard. This can be proved by transforming the CFCLP-TC to the well known uncapacitated facility location problem (UFLP). Since UFLP is NP-hard (Cornuejols et al. 1990), it follows that the CFCLP-TC is NP-hard, too.

5 True efficient sets

A multiobjective optimization problem is described by:

$$\min\big\{\big(f_1(x),\dots,f_p(x)\big):x\in X\big\},$$

where $f=(f_1,\dots,f_p)$ is the set of p real-valued objective functions, x is a solution to the problem and X is the set of feasible solutions.

One of the most popular approaches for generating efficient frontiers is the ε-constraint method. The ε-constraint method consists in a transformation of the problem by making constraints all except one objective as follows:

$$\min\big\{f_k(x):f_i(x)\le\varepsilon_i,i\ne k,x\in X\big\}.$$

The values of vectors ε_i are changed systematically to obtain the efficient frontier for the problem. Further details can be found in Steuer (1989) and Ehrgott (2005).

This method is preferred because some results in multiobjective optimization theory show that for combinatorial problems the weighted sum technique may not find all the efficient solutions (Ehrgott 2005). The complete set of efficient solutions for an instance is defined as the true efficient set. Recently, Ehrgott and Ruzika (2008) have proposed an improved version of the ε-constraint method that combines some ideas that address some weaknesses of the traditional method, namely the lack of easy-to-check conditions for properly efficient solutions and the inflexibility of the constraints.

In this work, three versions of the traditional ε-constraint method are developed. The procedures were coded in ANSI C. The single-objective subproblems of the ε-constraint method for fixed ε were solved by using the CPLEX 11.1 callable library. In our implementation of the ε-constraint methods, for enhancing the computational efficiency, we use the optimal solution found from a particular iteration when ε is fixed as an input for the IP subproblem associated with the following iteration. This has had a considerable positive impact on computational efficiency. For our specific application, it was empirically observed that an implementation based on the method by Ehrgott and Ruzika (2008) renders the same set of solutions as that obtained by the classical method with no benefit in terms of computational time.

5.1 Forward ε-constraint method (eC)

In this version of the ε-constraint method, function f_1 is used as the objective function and f_2 is used as a constraint. Therefore, the model described by (1)–(19) is rewritten as follows, keeping the same objective function (1):

MODEL 2:

$$\min f_1^t = \sum_{i \in I} \sum_{j \in J} \sum_{l \in LP_{ij}} CP_{ijl} X_{ijl} + \sum_{j \in J} \sum_{k \in K} \sum_{l \in LW_{jk}} CW_{jkl} Y_{jkl} + \sum_{j \in J} F_j Z_j$$

subject to $\quad T \le \varepsilon^t$ (21)

and constraints (3)–(19).

Each iteration of the algorithm is denoted by the index t. When the objective function (1) is optimized without constraint (21), it is expected that variable T takes its highest value. Thus it is natural to think of reducing the value of ε^t sequentially. With this logic, at each iteration the value of ε^t is reduced by a constant $\delta = 1$.

However, an optimal solution for this problem may result in constraint (21) being inactive. Hence it is convenient to recalculate the value of T from the values of A_{ijl} and B_{jkl} of that solution using (20). This strategy avoids solving problems with loose values of ε^t in constraint (21) for the next iteration of the algorithm. The algorithm is shown in Fig. 2. The exit condition in Step 4 implies that the value of T has a lower limit. Hence there is a point where the value of ε^t results in an infeasible problem.

5.2 Backward ε-constraint algorithm with lower and upper limits for f_2 (ReC-2B)

The branch-and-bound algorithm used by many commercial solvers allows using a known solution as incumbent solution to start the solution of a MIP. This utility im-

---

Algorithm eC

BEGIN
Input: Data instance of CFCLP-T.
Output: List of non-dominated solutions (*NDS*).

1. $NDS = \emptyset$, $t = 1$. Optimize Model 1 dropping objective function f_2.
2. Recalculate T using (20).
3. Register the solution in *NDS* with $f_1 = f_1^t$ and $f_2 = T$.
4. **while** (obtaining an optimal solution for Model 2)
    4.1. $t = t + 1$.
    4.2 Recalculate ε^t with $\delta = 1$:

$$\varepsilon^t = T - \delta$$

    4.3. Optimize Model 2.
    4.4. Recalculate T using (20).
    4.5. Register the solution in *NDS* with $f_1 = f_1^t$ and $f_2 = T$.
5. **endwhile**
6. Eliminate dominated solutions from *NDS*.
7. Return *NDS*.

END

Fig. 2 Pseudo-code for the forward ε-constraint algorithm (eC)

proves the efficiency of the branch-and-bound algorithm because many solutions can be discarded initially from the tree. Also this incumbent solution may help to produce useful cuts in the first stages of the branch-and-bound algorithm. We implemented this feature trying to reduce the run time of algorithm eC. This is possible when the following is observed. Consider two solutions S_1 and S_2 that are efficient solutions where $f_1(S_1) < f_1(S_2)$ and $f_2(S_1) > f_2(S_2)$. Thus, S_2 is a feasible solution for Model 2 when $\varepsilon^t = f_2(S_1)$. Hence S_2 can be used as the incumbent solution for that iteration of the ε-constraint method. This incumbent solution is used by the solver as initial solution and may reduce the run time in that iteration.

To use a starting solution, a modification to the algorithm has to be made. Instead of running the cycle reducing ε^t, now the value of this parameter has to be increased. Two issues have to be considered here. The value of T has to be recalculated at each iteration as well; however, this parameter is no longer useful for knowing the next value of ε^t in advance. Hence the movement of ε^t is always of $\delta = 1$ and some more weakly-efficient points are obtained. Also, the initial and final values of ε^t are not known unless a previous computation is made. To identify the lower limit of f_2, i.e., the initial value of ε^t, Model 1 must be optimized by dropping objective function f_1. To identify the upper limit of f_2, i.e., the final value of ε^t, Model 1 must be optimized by dropping objective function f_2 and recalculating the value of T with (20). The loops run between these extreme points. The algorithm is shown in Fig. 3.

5.3 Backward ε-constraint method with estimated lower limit for f_2 (ReC-1B)

The third version uses the same idea as that of the previous version, that is, using an existing solution as incumbent for the next iteration of the algorithm. It was observed that the linear relaxation of Model 1 dropping objective function f_1 was very loose, resulting in long times for Step 5 of the ReC-2B algorithm. One way to avoid this optimization was to estimate a lower limit for f_2 from the instance data and start the computation cycle from that point. The initial value of ε^t is estimated as follows:

$$\varepsilon^{\text{first}} = \min_j \left(\min_{i,l}(TP_{ijl}) + \min_{k,l}(TW_{jkl}) \right). \tag{22}$$

The implication of this procedure is that the cycle may begin from an infeasible problem for the initial value of ε^t, so that some extra computations may be made. The algorithm is a modification of that shown in Fig. 3. Steps 5 and 6 are substituted by a step in which $\varepsilon^{\text{first}}$ is estimated with (22). Steps 8.2, 8.3 and 8.4 are embedded in an *if* cycle that is executed only if a feasible solution is obtained from step 8.1.

6 Lower bound sets

Because of the computational complexity of the CFCLP-TC, relatively large instances of this problem may no longer be tractable from an exact optimization perspective. In that case, one uses heuristic methods that find an approximate set of non-dominated solutions. In that sense, having information about lower bounds for

Algorithm *ReC*-2*B*

BEGIN
Input: Data instance of CFCLP-TC.
Output: List of non-dominated solutions (*NDS*).

1. $NDS = \emptyset$, $t = 1$. Optimize Model 1 dropping objective function f_2.
2. Recalculate T using (20).
3. Register the solution in *NDS* with $f_1 = f_1^t$ and $f_2 = T$.
4. Register $\varepsilon^{\text{last}} = T$ as the final value for ε^t.
5. Optimize Model 1 dropping objective function f_1.
6. Register $\varepsilon^{\text{first}} = f_2$.
7. Initialize $t = 2$ and $\varepsilon^t = \varepsilon^{\text{first}}$.
8. **while** ($\varepsilon^t < \varepsilon^{\text{last}}$)
    8.1. Optimize Model 2.
    8.2. Recalculate T using (20).
    8.3. Register the solution in the *NDS* with $f_1 = f_1^t$ and $f_2 = T$.
    8.4. Register the solution as initial solution for the next iteration.
    8.5. $t = t + 1$.
    8.6 Recalculate ε^t with $\delta = 1$:

$$\varepsilon^t = \varepsilon^{t-1} + \delta$$

9. **endwhile**
10. Eliminate dominated solutions from *NDS*.
11. Return *NDS*.

END

Fig. 3 Pseudo-code for the Backward ε-constraint algorithm with lower and upper limits for f_2 (ReC-2B)

these non-dominated sets allows a measure of the quality of these solutions. In multi-objective problems, it is more appropriate to talk about a "lower bound set" (Ehrgott and Gandibleux 2007).

In this paper, several strategies based on linear relaxations within the branch-and-bound framework are proposed and tested. The idea is to attempt to exploit the fact that relaxing some sets of the variables may result in more tractable subproblems. The schemes are described next.

(a) Scheme LP (linear programming relaxation). It consists of relaxing all the integer variables. It is the most common method to obtain lower bounds. The quality of this bound is closely related to the formulation of the problem. In a branch-and-bound framework, this bound is equivalent to the LP relaxation computed at the root node.
(b) Scheme LPc (LP relaxation with cuts). It is basically the LP relaxation of the MIP followed by an effort to identify and add some common cuts. The addition of these cuts may in some cases strengthen considerable the LP relaxation. The following cuts used by default with CPLEX were Clique Cuts, Cover Cuts, Implied Bound Cuts, Flow Cuts. Flow Path Cuts, and Gomory Fractional Cuts.

(c) Scheme ABr (LP relaxation of variables A_{ijl} and B_{jkl}). These variables represent a big portion of the number of binary variables in large instances. By doing this relaxation, the resulting problem has as binary decision variables only those for selecting the distribution centers. Given that it is not a complete linear relaxation of the model, this lower bound should be better than the lower bound obtained through the scheme LPc.

(d) Scheme Zr (LP relaxation of variables Z_j). These variables are closely related to the facility location problem. Hence their relaxation may reduce the complexity of the problem, and a good lower bound may be obtained in shorter time. By this relaxation, the problem of facility location becomes a linear problem where variables Z_j take on fractional values and the flows in (8) are bounded by the value of $MW_j Z_j$. This scheme should have a better quality than the one obtained by scheme LPc.

For generating the complete lower bound set, each of these schemes is embedded within an algorithm similar to the ε-constraint method (ReC-1B) by fixing ε^t at each iteration using a fixed value of $\delta = 1$. Of course, in this algorithm, rather than solving the original MIP formulation, the focus is on solving the related relaxations.

7 Empirical evaluation

The specific goals accomplished by the experiments are as follows. The first goal is to evaluate the proposed exact algorithms to identify which of these is more efficient in obtaining the efficient frontier. The second goal is to present a detailed study of the profile of the efficient frontier, and to establish empirically that the two model objectives are indeed in conflict. Another goal was to identify the size of instances that can be solved in a reasonable time with the computing resources available. In terms of the lower bound sets, we wanted to identify the characteristics of the lower bound sets obtained through the schemes proposed and their efficiency in run time and quality.

To perform the computational study, instances are randomly generated as follows. For each instance, there are four main size parameters: the number of plants, the number of potential distribution centers, the number of customers, and the number of available transportation modes between each pair of nodes. The sizes generated are shown in Table 1, where the group code indicates: [number of plants—number of potential distribution centers—number of customers—number of available transportation modes between each pair of nodes]. The other parameters were randomly generated assuming some relations between these parameters as will be described below. Customer demands are drawn from a uniform distribution with values between 5000 and 10000. To guarantee to some extent that the single source constraint can be met, the capacities of the distribution centers must be higher than the maximum demand.

To avoid an "easy" facility location decision, some distribution centers must be able to supply the total demand. The assumption is that providing more options makes the instance harder. Hence the total demand D_T is the base to generate the distribution center capacity. The distribution center capacity is an integer random variable with

Table 1 Generated instances

Group code	Number of instances	Number of binary variables	Number of constraints
5-5-5-2	10	105	385
5-5-5-5	10	255	835
5-5-20-2	10	255	940
5-10-10-2	10	310	1115
5-10-15-2	10	410	1475
5-10-20-2	10	510	1835
5-5-20-5	10	630	2065

values between *MWlow* and *MWhigh* according to a uniform distribution, where these parameters are defined as follows:

$$D_T = \sum_{k \in K} D_k,$$

$$MWlow = \max_{k \in K}(D_k),$$

$$MWhigh = D_T + (D_T - MWlow) = 2D_T - MWlow.$$

The plant capacity must be generated also taking into consideration the total demand. In a feasible instance, the total capacity of the plants must satisfy the total demand. However, to generate a hard instance some plants are allowed to have a high capacity near to the total demand. The assumption is again that providing more options makes the problem harder. The plant capacity is an integer random variable with values between *MPlow* and *MPhigh* according to a uniform distribution, where these parameters are defined as shown below:

$$MPlow = \frac{D_T}{|I|}, \qquad MPhigh = D_T.$$

An assumption is that transportation cost and time are negatively correlated. For each arc, the time and cost are calculated but repeated parameter values are avoided for each pair of facilities. The transportation time is an integer random variable *Tarc* with values between 5 and 25 according to a uniform distribution. The unitary transportation cost *Carc* is a floating-point variable calculated with the *Tarc* value generated, as follows:

$$Carc = \frac{50}{Tarc}. \tag{23}$$

The generation of the distribution center fixed cost is based in some parameters already generated. An assumption is that the fixed cost is positively correlated with the distribution center capacity. To produce a hard instance, the total fixed cost must be close to the total transportation cost. As can be seen in (23), the maximum cost has a value of 10 which is used to estimate a reference transportation cost *Cref*. The average distribution center capacity *MWave* is calculated and used to compute the fixed

cost.

$$Cref = \frac{10 * D_T}{|J|}, \qquad MWave = \frac{\sum_{j \in J} MW_j}{|J|}, \qquad F_j = Cref * \frac{MW_j}{MWave}, \quad j \in J.$$

These are all the parameters required for an instance of the CFCLP-TC.

7.1 True efficient sets

The three algorithms previously presented were used to solve the generated instances. All procedures (exact algorithms and lower bounding schemes) were coded in C and compiled with Visual Studio 6.0. CPLEX 11.1 callable library (ILOG SA 2008) was used to solve optimally the sub-problems involved in the ε-constraint based algorithms (and in the lower bounding schemes also). These routines were run in a 3.0 GHz, 1.0 GB RAM, Intel Pentium 4 PC.

The instances of the groups 5-5-5-2, 5-5-5-5, 5-5-20-2, 5-10-10-2, and 5-10-15-2 were solved completely with the proposed algorithms, i.e., their complete true efficient sets were obtained. The results are shown in Table 2 comparing in the last columns the improvement in efficiency achieved by the algorithms ReC-2B and ReC-1B against the algorithm eC. The following is observed in these results:

- In 32 cases out of 50 (64%), ReC-2B was faster than eC.
  - Only for the favorable cases this improvement can be up to 23.9% of time reduction and 3.4% on the average.
- In 37 cases out of 50 (70%), ReC-1B was faster than eC.
  - Only for the favorable cases this improvement can be up to 26.6% of time reduction and 4.9% on the average.

Run times were similar between ReC-2B and ReC-1B, when individual iterations (each ε^t value) were compared. Yet, an overhead time is produced in ReC-2B when the initial value of ε^t is obtained through the optimization of f_2. Hence, a benefit is achieved in ReC-1B when the initial value of ε^t is obtained through the computation of a lower limit. Although some additional points had to be solved the time consumed is insignificant.

Figure 4 shows the efficient frontier for the instance 5-5-5-5-2. The efficient frontier for the rest of the mentioned instances is similar. The points are not connected because of the discretization of time units. The tradeoff between cost (f_1) and time (f_2) is evident.

Figure 5 shows the run times for the iterations of the eC algorithm in the 5-5-20-2-1 instance. This behavior is similar in all the instances. It is observed that intermediate values of ε^t create problems that are more difficult to solve, in general. This behavior can be explained as follows. First, as it is empirically shown in more detail in the following section, the initial gap in the branch-and-bound algorithm increases from high to low values of ε^t. This trend of the initial gap may be related to the increase of time to solve each MIP from high to intermediate values of ε^t. Nevertheless, when the value of ε^t is decreased the solution space becomes more constrained, and a reduced number of solutions must be explored. Also, for low values of ε^t CPLEX is more efficient in finding cuts to reduce the initial gap.

Table 2 Comparison of algorithms in terms of time usage

Group code	Instance	Time (CPU seconds)			Reduction in run time (%) vs. eC algorithm	
		eC	ReC-2B	ReC-1B	ReC-2B	ReC-1B
5-5-5-2	1	9.4	7.6	7.1	18.7	24.7
	2	15.0	11.4	11.0	23.9	26.6
	3	10.8	13.0	11.7	−19.6	−7.9
	4	30.0	25.4	24.3	15.5	18.9
	5	10.3	10.4	10.3	−1.5	−0.6
	6	22.3	20.7	20.6	7.1	7.8
	7	15.2	11.6	11.3	23.8	25.3
	8	15.9	12.4	12.1	22.1	24.1
	9	13.8	10.6	10.4	23.2	24.4
	10	24.7	24.7	22.8	−0.1	8.0
5-5-5-5	1	143.3	139.0	134.9	3.0	5.8
	2	181.3	158.0	157.8	12.8	12.9
	3	227.4	219.2	219.2	3.6	3.6
	4	188.2	179.8	179.8	4.5	4.5
	5	120.9	111.2	111.1	8.0	8.1
	6	109.6	102.3	101.5	6.7	7.4
	7	123.9	118.0	117.2	4.8	5.4
	8	149.4	134.7	134.0	9.8	10.4
	9	158.8	146.3	145.6	7.9	8.3
	10	174.1	151.0	150.2	13.3	13.7
5-5-20-2	1	279.5	235.8	235.5	15.7	15.7
	2	254.8	269.2	268.6	−5.6	−5.4
	3	446.5	455.8	452.2	−2.1	−1.3
	4	290.5	324.7	323.8	−11.8	−11.4
	5	396.8	496.2	491.0	−25.1	−23.7
	6	286.7	302.4	301.3	−5.5	−5.1
	7	478.3	503.0	499.8	−5.2	−4.5
	8	426.8	493.8	492.8	−15.7	−15.4
	9	322.4	322.9	321.5	−0.2	0.3
	10	409.1	430.5	430.3	−5.2	−5.2
5-10-10-2	1	860.1	834.7	834.4	3.0	3.0
	2	507.2	500.1	500.1	1.4	1.4
	3	1101.7	963.6	960.7	12.5	12.8
	4	646.9	593.9	583.2	8.2	9.9
	5	638.8	633.6	631.7	0.8	1.1
	6	1011.0	1029.5	1019.9	−1.8	−0.9
	7	672.3	592.1	585.6	11.9	12.9
	8	555.4	592.4	582.2	−6.7	−4.8

Table 2 (*Continued*)

| Group code	Instance	Time (CPU seconds)			Reduction in run time (%) vs. eC algorithm	
		eC	ReC-2B	ReC-1B	ReC-2B	ReC-1B
	9	597.9	550.7	550.7	7.9	7.9
	10	791.7	690.6	682.3	12.8	13.8
5-10-15-2	1	15131.2	13889.4	13836.3	8.2	8.6
	2	23555.9	22136.2	21820.5	6.0	7.4
	3	25971.3	30189.9	29860.6	−16.2	−15.0
	4	37599.2	37060.1	36416.7	1.4	3.1
	5	21181.8	19518.4	19350.4	7.9	8.6
	6	24924.8	28100.7	26623.9	−12.7	−6.8
	7	33744.1	34819.5	32181.9	−3.2	4.6
	8	29886.3	27506.5	27337.4	8.0	8.5
	9	20102.2	22378.8	22253.2	−11.3	−10.7
	10	15745.5	14863.2	14753.2	5.6	6.3
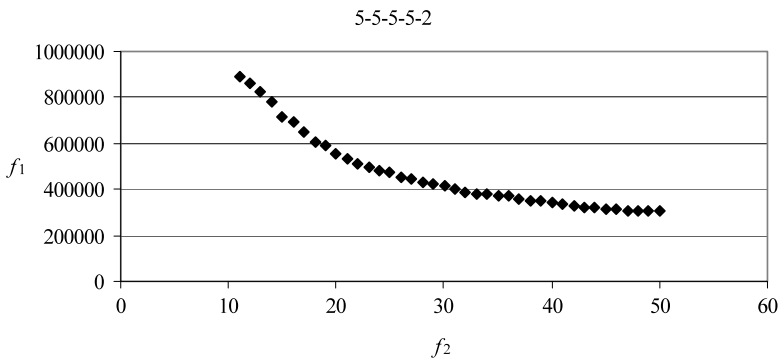
Fig. 4 Set of non-dominated solutions for the instance 5-5-5-5-2

7.2 Lower bound sets

The goals of these experiments are to study the quality of the lower bound sets obtained with the proposed schemes, and to gain insight in the structure of the problem. Four lower bounding schemes based on linear relaxations of the MIP were evaluated. These are described in Sect. 6. In summary, scheme LP is the linear programming relaxation of the MIP. Scheme LPc is the LP relaxation with CPLEX cuts at the end of Node 0. Scheme ABr is the LP relaxation of the variables A_{ijl} and B_{jkl}. Finally, scheme Zr is the LP relaxation of variables Z_j. For generating the lower bound sets in a specific instance, the lower bounding schemes were used within the framework of ReC-1B algorithm with some modifications about fixing the initial and last values of the parameter ε^t according to the instance data. This procedure was coded in C
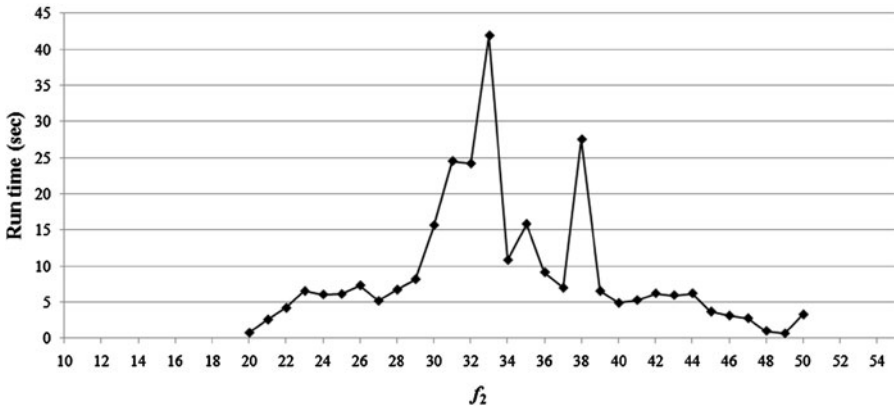
Fig. 5 Run time behavior of eC algorithm as a function of the value of ε^t
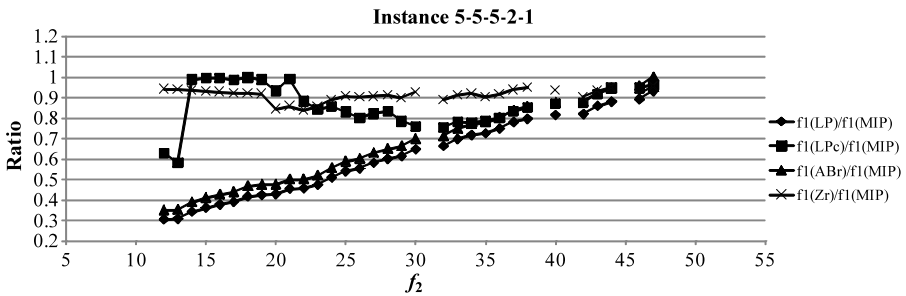

Fig. 6 Quality of the lower bound sets for instance 5-5-5-2-1

and compiled with Visual Studio 6.0. CPLEX 11.1 callable library (ILOG SA 2008) was used to solve optimally the sub-problems. The routines were run in a 3.0 GHz, 1.0 GB RAM, Intel Pentium 4 PC.

To define the quality of the lower bound set the schemes were tested in five instances of each of the following data sets, 5-5-5-2, 5-5-5-5, 5-5-20-2, 5-10-10-2, and 5-10-15-2. For a fixed value of ε^t, the ratio f_1 (scheme)/f_1 (MIP) was calculated. The behavior of these ratios is shown in Fig. 6 for instance 5-5-5-2-1. Figures 7 and 8 in the Appendix show he behavior for instances 5-5-5-5-1 and 5-5-20-2-1, respectively. This trend is similar for the other instances examined.

Table 3 in the Appendix presents the average, minimum and maximum values of the f_1 (scheme)/f_1 (MIP) ratios for the lower bound schemes proposed. Data for scheme Zr and instances 5-10-10-2 and 5-10-15-2 are not included in Table 3 because the runs were stopped after a limit of 10 hours and only partial information is available.

It is evident from the results that the best lower bound set is obtained with the scheme Zr that uses the linear relaxation of variables Z_j. In order of quality the second best is the LPc scheme. The third best is obtained with the ABr scheme through the linear relaxation of variables A_{ijl} and B_{jkl}. The worst lower bound set is obtained with the complete linear relaxation of the MIP model, i.e., the LP scheme.

A degradation of the lower bound set is observed for the lowest values of ε^t in the ABr and LP schemes. It is because the value of ε^t is loose in constraint (21) and the value of f_1 barely changes between the lowest and highest values of ε^t, while the value of f_1 in the MIP increases for lower values of ε^t. The run times for the linear relaxation schemes are displayed in Table 4 in the Appendix. The column MIP represents the time required by the ReC-1B algorithm to obtain the optimal solution.

In the case of the LPc scheme, a recovery can be observed in the lowest and highest values of ε^t. The reason for the improvement of these ratios may be a more efficient use of the solver default heuristics and cuts for strengthening the linear relaxation of the MIP. Although the quality of the lower bound set with the Zr scheme is high, the run time is a disadvantage because it is longer than the time used to solve the CFCLP-TC with the ReC-1B algorithm. This behavior is counterintuitive because it is expected that a linear relaxation may be solved in less time than the MIP model. There is one reason for this behavior. The magnitude of the cost associated with variables Z_j in the objective function f_1 is high. When these variables are continuous CPLEX is unable to find useful cuts with a high impact in the gap. Therefore, the gap is not improved fast enough during the exploration of the nodes in the branch-and-bound algorithm.

8 Conclusions

In the process of supply chain design, many decisions have to be made and several aspects must be taken into account. An area of opportunity was identified by incorporating the selection of the transportation mode in the supply chain network design while generating a set of efficient solutions for the decision maker in terms of the cost–time tradeoff. The problem introduced here considers the discrete facility location in the intermediate echelon of a two-echelon distribution network, the selection of the transportation modes between facilities, and the determination of the flows between facilities. We have named this new problem as the "Capacitated Fixed Cost Facility Location Problem with Transportation Choices" (CFCLP-TC). To the best of our knowledge this problem has not been addressed before.

To solve the model, three versions (named eC, ReC-2B, and ReC-1B) of the classical ε-constraint method were studied. Within the ε-constraint method, the cost function was considered as objective function while the time function was handled as a constraint with changing right-hand side values (ε^t). These algorithms were designed to obtain the set of efficient solutions for a problem instance.

The numerical results showed that the ReC-1B algorithm was more efficient (faster) than algorithms ReC-2B and eC. The ReC-1B algorithm takes advantage of using an initial solution in each optimization step and obtains the initial value of ε^t from a lower limit estimated with the instance data. We infer that the ReC-1B algorithm may be adapted to other bi-objective problems easily. Therefore, we see a contribution in showing clearly a way to implement efficiently the ε-constraint method taking advantage of the setting parameters available in commercial optimizers.

In addition, four different strategies for obtaining lower bound sets were examined. These were based on linear relaxations of the MIP model, named LP, LPc, ABr, and Zr. The best lower bound sets in terms of quality were obtained with the Zr scheme with a more homogeneous behavior along the efficient frontier. The main disadvantage is its relatively large run time. Further work must be done to obtain lower bound sets of high quality for this problem in reasonable run times. An option is to attempt to speed up scheme Zr by means of a Lagrangian relaxation scheme or relaxing the single-sourcing constraint.

Future work related to possible model extensions include the case of multiple commodities, the use of direct flows from plants to customers, considering flows between distribution centers, and including the processing times in the plants and distribution centers. Also a third objective may be included to minimize total transportation time of the system. The interdependency of the choice on the transportation mode with strategic inventory decisions and international supply chain aspects is an important aspect that needs to be further addressed. Some other issues worthwhile studying are the variability of some of the model parameters, which would yield a stochastic program, the decisions on the inventory levels at the distribution centers, or the integration with routing decisions.

The proposed solution procedures are a good alternative for small- to medium-size instances; however, it is clear they are not suitable for large-scale instances that may represent a number of real-world problems due to the inherent NP-completeness of the problem. Therefore, it is evident that the computational complexity of the CFCLP-TC makes necessary the use of a heuristic method to solve larger instances. This conclusion remains true for the extensions proposed for the model.
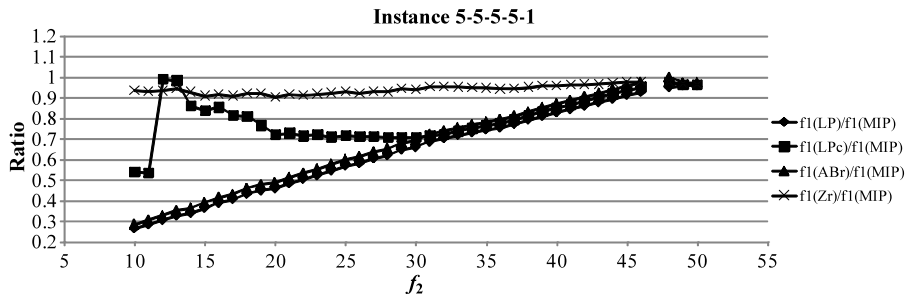
Appendix

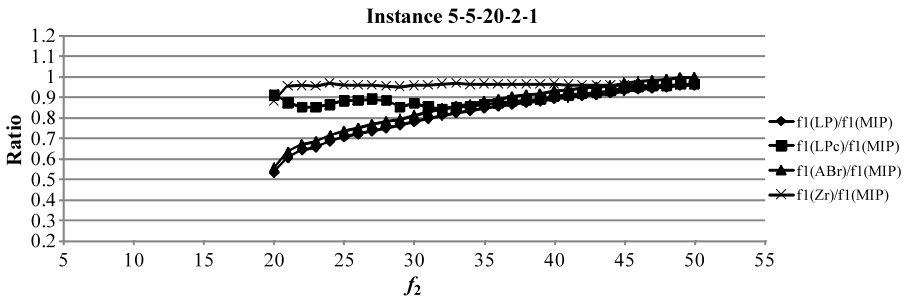**Fig. 7** Quality of the lower bound sets for instance 5-5-5-5-1

Fig. 8 Quality of the lower bound sets for instance 5-5-20-2-1

Table 3 Comparison of f_1(scheme)/f_1 (MIP) ratios for the lower bounding schemes

Group code	Ins.	Average ratio f_1 (scheme)/f_1 (MIP)				Minimum ratio f_1 (scheme)/f_1 (MIP)				Maximum ratio f_1 (scheme)/f_1 (MIP)			
		LP	LPc	ABr	Zr	LP	LPc	ABr	Zr	LP	LPc	ABr	Zr
5-5-5-2	1	0.60	0.86	0.65	0.91	0.30	0.58	0.35	0.84	0.93	1.00	1.00	0.96
	2	0.68	0.85	0.76	0.92	0.38	0.74	0.42	0.87	0.90	0.97	1.00	0.96
	3	0.61	0.84	0.73	0.89	0.33	0.77	0.40	0.84	0.83	1.00	0.99	0.94
	4	0.63	0.82	0.71	0.86	0.33	0.67	0.39	0.81	0.89	1.00	1.00	0.91
	5	0.61	0.85	0.67	0.87	0.35	0.73	0.41	0.83	0.92	0.99	1.00	0.93
5-5-5-5	1	0.62	0.79	0.66	0.94	0.27	0.54	0.29	0.90	0.95	0.99	1.00	0.98
	2	0.68	0.82	0.72	0.95	0.32	0.56	0.35	0.91	0.95	1.00	1.00	0.97
	3	0.64	0.79	0.73	0.90	0.28	0.65	0.32	0.85	0.88	0.98	1.00	0.94
	4	0.64	0.78	0.68	0.91	0.29	0.66	0.31	0.88	0.94	1.00	1.00	0.95
	5	0.63	0.80	0.65	0.91	0.23	0.65	0.26	0.78	0.96	1.00	1.00	0.96
5-5-20-2	1	0.82	0.90	0.85	0.96	0.54	0.84	0.56	0.88	0.96	0.96	1.00	0.97
	2	0.80	0.88	0.83	0.96	0.53	0.81	0.55	0.92	0.96	0.97	1.00	0.98
	3	0.78	0.87	0.80	0.95	0.52	0.79	0.53	0.90	0.98	0.98	1.00	0.98
	4	0.78	0.85	0.81	0.96	0.51	0.76	0.53	0.93	0.97	0.97	1.00	0.98
	5	0.77	0.85	0.81	0.94	0.49	0.79	0.51	0.89	0.96	0.97	1.00	0.97
5-10-10-2	1	0.71	0.85	0.72	–	0.33	0.76	0.35	–	0.97	1.00	0.99	–
	2	0.68	0.86	0.70	–	0.31	0.73	0.32	–	0.96	1.00	0.98	–
	3	0.70	0.85	0.71	–	0.33	0.74	0.34	–	0.97	1.00	0.99	–
	4	0.67	0.83	0.68	–	0.28	0.52	0.29	–	0.96	1.00	0.97	–
	5	0.71	0.84	0.72	–	0.36	0.75	0.37	–	0.96	0.97	0.98	–
5-10-15-2	1	0.77	0.87	0.78	–	0.45	0.80	0.46	–	0.96	0.97	0.97	–
	2	0.74	0.85	0.75	–	0.40	0.76	0.41	–	0.97	0.98	0.99	–
	3	0.76	0.85	0.76	–	0.46	0.77	0.46	–	0.97	0.98	0.98	–
	4	0.73	0.84	0.74	–	0.38	0.75	0.38	–	0.97	0.98	0.99	–
	5	0.71	0.86	0.72	–	0.35	0.77	0.36	–	0.97	1.00	0.99	–

Table 4 Run time (CPU seconds) for the lower bounding schemes

Group code	Instance	Lower bounding schemes				
		MIP	LP	LPc	ABr	Zr
5-5-5-2	1	7.1	0.3	1.4	3.5	24.8
	2	11.0	0.3	1.4	2.4	25.1
	3	11.7	0.3	0.5	2.4	26.1
	4	24.3	0.3	0.7	4.5	39.8
	5	10.3	0.3	0.7	3.8	32.6
5-5-5-5	1	134.9	0.6	1.4	5.2	4275.7
	2	157.8	0.6	1.1	1.8	6032.6
	3	219.2	0.8	1.3	2.7	6792.8
	4	179.8	1.2	0.7	5.4	5026.7
	5	111.1	0.6	0.9	3.2	5113.5
5-5-20-2	1	235.5	1.0	1.9	7.5	382.4
	2	268.6	0.9	1.2	6.2	525.8
	3	452.2	1.1	0.8	8.1	985.1
	4	323.8	0.9	0.9	10.8	725.8
	5	491.0	0.9	1.0	14.0	960.3
5-10-10-2	1	834.4	1.0	2.7	23.2	–
	2	500.1	0.9	2.1	21.8	–
	3	960.7	1.1	2.1	35.1	–
	4	583.2	1.0	2.1	17.0	–
	5	631.7	1.0	1.9	31.1	–
5-10-15-2	1	13836.3	1.6	2.3	48.5	–
	2	21820.5	1.4	3.8	69.5	–
	3	29860.6	1.5	2.2	20.5	–
	4	36416.7	1.6	3.3	35.3	–
	5	19350.4	1.5	1.6	48.3	–

References

Aikens CH (1985) Facility location models for distribution planning. Eur J Oper Res 22(3):263–279

Altiparmak F, Gen M, Lin L, Paksoy T (2006) A genetic algorithm approach for multi-objective optimization of supply chain networks. Comput Indust Eng 51(1):197–216

Arntzen BC, Brown GC, Harrison TP, Trafton LL (1995) Global supply chain management at Digital Equipment Corporation. Interfaces 25(1):69–93

Ballou RH (1999) Business logistics management. Prentice Hall, New York

Baumol WJ, Wolfe P (1958) A warehouse–location problem. Oper Res 6(2):252–263

Benjamin J (1990) An analysis of mode choice for shippers in a constrained network with applications to just-in-time inventory. Transp Res Part B 24(3):229–245

Chan FTS, Chung SH, Choy KL (2006) Optimization of order fulfillment in distribution network problems. J Intell Manuf 17(3):307–319

Chopra S, Meindl P (2004) Supply chain management: strategy, planning, and operation. Prentice Hall, New York

Cordeau JF, Pasin F, Solomon MM (2006) An integrated model for logistics network design. Ann Oper Res 144(1):59–82

Cornuejols G, Nemhauser GL, Wolsey LA (1990) The uncapacitated facility location problem. In: Mirchandani PB, Francis RL (eds) Discrete location theory. Wiley, New York, pp 119–171, Chap 3

Current J, Min H, Schilling D (1990) Multiobjective analysis of facility location decisions. Eur J Oper Res 49(3):295–307

Ehrgott M (2005) Multicriteria optimization. Springer, Berlin

Ehrgott M, Gandibleux X (2007) Bound sets for biobjective combinatorial optimization problems. Comput Oper Res 34(9):2674–2694

Ehrgott M, Ruzika S (2008) Improved ε-constraint method for multiobjective programming. J Optim Theory Appl 138(3):375–396

ElMaraghy HA, Majety R (2008) Integrated supply chain design using multi-criteria optimization. Int J Adv Manuf Technol 37(3):371–399

Eskigun E, Uzsoy R, Preckel PV, Beaujon G, Krishnan S, Tew JD (2005) Outbound supply chain network design with mode selection, lead times and capacitated vehicle distribution centers. Eur J Oper Res 165(1):182–206

Farahani RZ, SteadieSeifi M, Asgari N (2009) Multiple criteria facility location problems: a survey. Appl Math Modell 34(7):1689–1709

Graves SC, Willems SP (2005) Optimizing the supply chain configuration for new products. Manag Sci 51(8):1165–1180

ILOG SA (2008) ILOG CPLEX Callable Library C API 11.1 Reference Manual. ILOG, France

Klose A, Drexl A (2005) Facility location models for distribution system design. Eur J Oper Res 162(1): 4–29

Kuehn AA, Hamburger MJ (1963) A heuristic program for locating warehouses. Manag Sci 9(4):643–666

Melo MT, Nickel S, Saldanha-da-Gama F (2009) Facility location and supply chain management—a review. Eur J Oper Res 196(2):401–412

Sahin G, Sural H (2007) A review of hierarchical facility location models. Comput Oper Res 34(8):2310–2331

Simchi-Levi D, Kaminski P, Simchi-Levi E (2000) Designing and managing the supply chain: concepts, strategies and case studies. McGraw Hill, New York

Steuer RE (1989) Multiple criteria optimization: theory, computation and application. Krieger, Melbourne

Thomas DJ, Griffin PM (1996) Coordinated supply chain management. Eur J Oper Res 94(1):1–15

Truong TH, Azadivar F (2005) Optimal design methodologies for configuration of supply chains. Int J Prod Res 43(11):2217–2236

Vidal CJ, Goetschalckx M (1997) Strategic production–distribution models: a critical review with emphasis on global supply chain models. Eur J Oper Res 98(1):1–18

Vidyarthi N, Elhedhli S, Jewkes E (2009) Response time reduction in make-to-order and assemble-to-order supply chain design. IIE Trans 41(5):448–466

Wilhelm W, Liang D, Rao B, Warrier D, Zhu X, Bulusu S (2005) Design of international assembly systems and their supply chains under NAFTA. Transp Res Part E 41(6):467–493

Zeng DD (1998) Multi-issue decision making in supply chain management and electronic commerce. PhD dissertation, Graduate School of Industrial Administration and Robotics Institute, Carnegie Mellon University, Pittsburgh, USA

Zhou G, Min H, Gen M (2003) A genetic algorithm approach to the bi-criteria allocation of customers to warehouses. Int J Prod Econ 86(1):35–45