

A Supply Chain Problem with Facility Location and Bi-objective Transportation Choices

Elias Olivares-Benitez
School of Engineering
Tecnologico de Monterrey
Aguascalientes, México
Elias.Olivares@itesm.mx

José Luis González-Velarde
Tecnologico de Monterrey
Monterrey, México
Gonzalez.Velarde@itesm.mx

Roger Z. Ríos-Mercado
Graduate Program in Systems Engineering
Universidad Autónoma de Nuevo León
San Nicolás de los Garza, México
roger@mail.uanl.mx

22 October 2008

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 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 this model for this new bi-objective optimization problem, and a computational study of implementations of both the ε -constrained methods for obtaining optimal efficient fronts, and the lower bounding schemes.

Keywords: supply chain design, facility location, multi-objective, lead time, transportation channel, lower bound set

1 Introduction

Supply chains have received much attention recently after recognizing the importance of the logistic costs in the cost structure of the products. Their efficient management provides a competitive advantage to domestic and international firms. At the strategic level the managers must design the supply chain to achieve the minimum cost and to meet a level of customer service. This task concerns aspects of inventory, transportation and facility location (Ballou, 1999). The supply chain, also known as distribution network 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, Kaminsky, and Simchi-Levi, 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 logistic 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. A high number of facilities may reduce the lead 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 location of facilities 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 Drexel, 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 often is 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 logistic 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 Section 2. Section 3 shows the problem description in detail. The mathematical framework is explained in Section 4. Section 5 is dedicated to present 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 Section 6. These schemes are based on linear relaxations of the model and on the information of the 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 Section 8.

2 Literature Review

Several reviews about models for supply chain design (Aikens, 1985; Thomas and Griffin, 1996; Vidal and Goetschalckx, 1997; Klose and Drexel, 2005) 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, Min, and Schilling (1990) it is evident that the balance of these measures has not been studied extensively. Some relevant works produced recently are highlighted next.

2.1 Cost-time tradeoff with one transportation mode between facilities

A first set of works include 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, Min, and Gen (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 inter-modal 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. Altıparmak 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. Selection of 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.

2.2 Multiple transportation modes between facilities without cost-time tradeoff

A second set of papers appreciate the influence of the transportation channel selection 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 Benders decomposition technique. Wilhelm et al. (2005) also include transportation mode selection 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, Pasin, and Solomon (2006) presented a model that has many of the elements of classic distribution network design. Additionally, the selection of 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 used a Benders decomposition method, commonly used in these types 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, Chung, and Choy (2006) present a multi-objective 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 by 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.

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 multi-objective 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 can not 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 can not 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 states that each distribution center is supplied at most by one distribution center, what is called a single source constraint. However, the demand of each customer must be met.

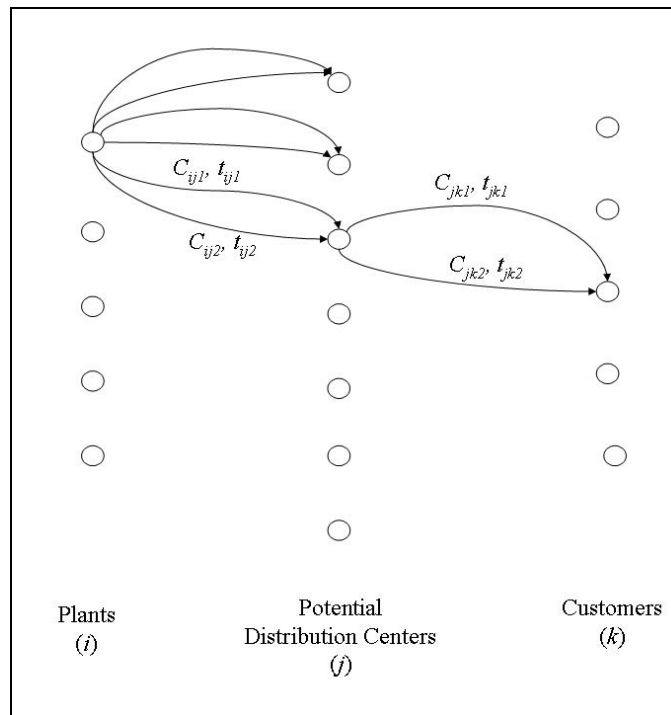


Figure 1. Single product, single period, and two-echelon distribution system. Each transportation channel has a time and a unitary cost associated.

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 logistic 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 cost, i.e. a faster service will be more expensive. A scheme of the distribution network is shown in [Figure 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 in 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. 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 nodes origin-destination 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 in 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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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 sum of the maximum lead time from the plants to the customers through each distribution center. This function was reformulated from equation (20) to eliminate the non-linearity:

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

Constraints (3)-(5) complete the linearization of equation (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, equation (20) considers the paths through open distribution centers and minimizes the maximum time along the supply chain. Constraint (6) requires the demand satisfaction of each customer. Constraint (7) is formulated for not exceeding the capacity limits of the plants. Constraint (8) states 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. Constraint (9) keeps the flow balance at the distribution center. Constraint (10) establishes 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). Constraint (17) ensures 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. Constraint (18) is for continuous non negative variables. Binary variables are required in constraint (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 fixed the values for A_{ijl} , B_{jkl} and Z_j , the remaining structure is a transportation problem. It is well known that the unimodularity property of this problem produce these integral results under such condition.

It should be noted that constraint (10) implies 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.

4.2 Computational complexity

Theorem: *The CFCLP-TC is NP-hard.*

Proof: To prove this, we first observe that given that there is a polynomial number of both variables and constraints, feasibility of any solution can be checked in polynomial time. The second part of the proof consists of showing that the well-known UFLP, which is known to be NP-hard, can be polynomially reduced to CFCLP-TC. The main idea is to consider a special case that disregards the decisions in the first echelon of the supply chain. Let us consider a special case of CFCLP-TC. First, let $|LP_{ij}| = |LW_{ij}| = 1$ and $TP_{ijl} = TW_{jkl} = 0$. This implies that subindex l can be dropped from the formulation, that time-related constraints (3)-(5) become

redundant, and objective f_2 becomes constant (=0) and can thus be dropped from the problem. This also implies that (10)-(12) and (17) are always satisfied. Then, by letting $MP_i = M$ (relaxing the plant capacity), where M is a very large number, constraint (7) becomes redundant. It is clear to see that, by considering an instance with $CP_{ijl} = 0$, the first term in the objective function reduces to zero, and constraints (9) and (13)-(16) become redundant. Finally, by letting $MW_i = M$ (relaxing the distribution center capacity), and redefining Y_{jk} as the fraction of the demand required by customer k supplied by distribution center j , the demand parameter D_k is moved from constraint (6) to objective function (1), and constraints (6) and (8) are rewritten to obtain the following model:

$$\min f_1 = \sum_{j \in J} \sum_{k \in K} D_k CW_{jk} Y_{jk} + \sum_{j \in J} F_j Z_j$$

subject to

$$\begin{aligned} \sum_{j \in J} Y_{jk} &= 1 & k \in K \\ Z_j - Y_{jk} &\geq 0 & j \in J, k \in K \\ Y_{jk} &\geq 0 & j \in J, k \in K \\ Z_j &\in \{0,1\} & j \in J \end{aligned}$$

This is the uncapacitated facility location problem (UFLP). Since UFLP is NP-hard (Cornuejols, Nemhauser, and Wolsey, 1990), it follows CFCLP-TC is NP-hard too and the proof is complete. ■

5 True Efficient Sets

One of the most popular approaches for generating efficient frontiers is the ε -constraint method (Steuer, 1989; Ehrgott, 2005). This method is preferred because some results in multi-objective 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. In this work, three versions of the ε -constraint method are developed. These algorithms try to take advantage of the use of an incumbent solution in the branch and bound algorithm used by any commercial solver for MIP problems.

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

$$T \leq \varepsilon^t \tag{21}$$

and constraints (3)-(19).

The index t is used for the iteration into the algorithm. 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 $\delta=1$.

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 equation (22).
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 equation (22).
 - 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

Figure 2. Pseudocode for the forward ε -constraint algorithm (eC).

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 equation (22):

$$T = \max_j \left(\max_{i,l} (TP_{ijl} A_{ijl}) + \max_{k,l} (TW_{jkl} B_{jkl}) \right) \quad (22)$$

This strategy avoids solving problems with loose values of ε^l in constraint (21) for the next iteration of the algorithm. The algorithm is shown in [Figure 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^l 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 improves 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 $\varepsilon^l = f_2(S_1)$. 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 implement the starting solution tool a modification to the algorithm has to be made. Instead of running the cycle reducing ε^l , 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 in each iteration as well; however, this parameter is no longer useful for knowing the next value of ε^l in advance. Hence the movement of ε^l is always of $\delta = 1$ and some more weakly-efficient points are obtained. Also, the initial and final values of ε^l are not known unless a previous computation is made. To identify the lower limit of f_2 , i.e. the initial value of ε^l , Model 1 must be optimized dropping objective function f_1 . To identify the upper limit of f_2 , i.e. the final value of ε^l , Model 1 must be optimized dropping objective function f_2 and recalculating the value of T with equation (22). The loops run between these extreme points. The algorithm is shown in [Figure 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^l is estimated as follows:

Algorithm *ReC-2B*

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 equation (22).
3. Register the solution in *NDS* with $f_1 = f_1^t$ and $f_2 = T$.
4. Register $\varepsilon^{last} = T$ as the final value for ε^t .
5. Optimize Model 1 dropping objective function f_1 .
6. Register $\varepsilon^{first} = f_2$.
7. Initialize $t = 2$ and $\varepsilon^t = \varepsilon^{first}$.
8. **while** ($\varepsilon^t < \varepsilon^{last}$)
 - 8.1. Optimize Model 2.
 - 8.2. Recalculate T using equation (22).
 - 8.3. Register the solution in the *NDS* with $f_1 = f_1^t$ and $f_2 = T$.
 - 8.4. Register the solution as start 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

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

$$\varepsilon^{first} = \min_j \left(\min_{i,l} (TP_{ijl}) + \min_{k,l} (TW_{jkl}) \right) \quad (23)$$

The implication of this procedure is that the cycle may begin from an infeasible problem for the start value of ε^t , so that some extra computations may be made. The algorithm is shown in **Figure 4**.

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 sorts to heuristic methods that find an approximate set of non-dominated solutions. In that sense, having information about lower bounds for these non-dominated sets allows a measure

of the quality of these solutions. In multiobjective problems it is more appropriate to talk about a “lower bound set” (Ehrgott and Gandibleux, 2007).

Algorithm *ReC-1B*

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 equation (22).
3. Register the solution in the *NDS* with $f_1 = f_1^t$ and $f_2 = T$.
4. Register $\varepsilon^{last} = T$ as the final value for ε^t .
5. Estimate the value ε^{first} with equation (23).
6. Initialize $t = 2$ and $\varepsilon^t = \varepsilon^{first}$.
7. **while** ($\varepsilon^t < \varepsilon^{last}$)
 - 7.1. Optimize Model 2.
 - 7.2. **if** (a feasible solution is obtained)
 - 7.2.1. Recalculate T using equation (22).
 - 7.2.2. Register the solution in the *NDS* with $f_1 = f_1^t$ and $f_2 = T$.
 - 7.2.3. Register the solution as start solution for the next iteration.
 - 7.3. **endif**
 - 7.4. $t = t + 1$.
 - 7.5. Recalculate ε^t with $\delta=1$:
$$\varepsilon^t = \varepsilon^{t-1} + \delta$$
8. **endwhile**
9. Eliminate dominated solutions from *NDS*.
10. Return *NDS*.

END

Figure 4. Pseudocode for the Backward ε -constraint algorithm with estimated lower limit for f_2 (ReC-1B).

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 considerably 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 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 equation (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 are embedded within an algorithm similar to the ε -constrained method (ReC-1B) by fixing ε' 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. First, to evaluate the proposed exact algorithms to identify which of these is more efficient in obtaining the efficient frontier. Then, 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, 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 arcs between 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 arcs between nodes]. The other parameters were randomly generated assuming some relations between these parameters. The customer demand is an integer random variable with values between 5000 and 10000 according to a uniform distribution. 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.

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
5-10-20-2	10	510	1835
5-5-20-5	10	630	2065

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 values between $MWlow$ and $MWhigh$ with 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 taking into consideration the total demand also. 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 $MPhigh$ with a uniform distribution, where these parameters are defined as shown below:

$$MPlow = \frac{D_T}{|I|}$$

$$MPhigh = D_T$$

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 with 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 equation (23) the maximum cost has a value of 10 which is used to estimate a reference transportation cost C_{ref} . The average distribution center capacity MW_{ave} is calculated and used to compute the fixed cost.

$$C_{ref} = \frac{10 * D_T}{|J|}$$

$$MW_{ave} = \frac{\sum_{j \in J} MW_j}{|J|}$$

$$F_j = C_{ref} * \frac{MW_j}{MW_{ave}} \quad j \in J$$

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 9.1 callable library (ILOG SA, 2005) 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 and 5-5-20-2 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 10 cases out of 30 (33%) ReC-2B was faster than eC.
 - Only for the favorable cases this improvement can be up to 56.1% of time reduction and 24.1% in the average.
- In 26 cases out of 30 (87%) ReC-1B was faster than eC.
 - Only for the favorable cases this improvement can be up to 59.5% of time reduction and 19% in the average.

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

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	6.6	7.5	6.3	-	4.7
	2	12.0	13.6	11.2	-	6.5
	3	11.2	12.5	10.4	-	7.8
	4	20.5	18.6	17.7	9.3	13.9
	5	9.8	11.1	9.5	-	3.7
	6	20.1	19.8	19.3	1.6	4.0
	7	12.9	15.2	11.6	-	9.8
	8	12.2	17.9	11.4	-	6.8
	9	12.1	12.5	10.3	-	14.6
	10	34.8	35.0	28.8	-	17.1
5-5-5-5	1	152.7	164.5	141.3	-	7.4
	2	343.0	337.2	334.8	1.7	2.4
	3	402.2	2551.4	346.5	-	13.8
	4	442.2	808.1	359.2	-	18.8
	5	159.5	246.8	139.0	-	12.8
	6	193.2	430.2	161.6	-	16.4
	7	175.6	321.8	176.3	-	-
	8	254.0	829.1	182.7	-	28.1
	9	536.3	1052.8	452.8	-	15.6
	10	1722.4	1651.9	1444.3	4.1	16.1
5-5-20-2	1	1976.0	2614.8	1981.9	-	-
	2	4096.4	1975.3	1658.1	51.8	59.5
	3	10466.4	12745.5	12709.4	-	-
	4	4350.8	2637.9	2436.2	39.4	44.0
	5	7161.7	3776.4	3676.3	47.3	48.7
	6	3627.2	1590.6	1598.0	56.1	55.9
	7	8513.4	7786.1	7727.7	8.5	9.2
	8	12117.6	9567.5	9456.2	21.0	22.0
	9	2896.1	3338.9	2973.0	-	-
	10	9884.9	10327.8	6573.0	-	33.5

The groups of instances 5-10-10-2, 5-10-20-2 and 5-5-20-5 were not solved completely because the computer ran out of memory during the process. The instances of the group 5-10-10-2 were solved with a time limit of 3600 seconds (*TILIM*) to reinforce the conclusions about the efficiency of the algorithms. Given the previous results, only the eC and ReC-1B algorithms were tested for these instances. Here, in both algorithms several iterations reached the time limit and thus the individual gaps (%) were compared for those points. **Table 3** displays the results for these instances. Columns 5 and 6 show the number of iterations where *TILIM* was reached before obtaining the optimal solution. Column 7 shows the number of iterations where *TILIM* was reached before obtaining the optimal solution in common for both algorithms. Column 8 compares the number of iterations where *TILIM* was reached before obtaining the optimal solution with the ReC-1B algorithm and with a better Gap compared to the eC algorithm. Finally, Column 9 shows the average gap (%) for both algorithms where *TILIM* was reached before obtaining the optimal solution.

Table 3. Evaluation of eC and ReC-1B for the largest set of instances (group 5-10-10-2) with a time limit of 3600 sec.

Ins.	Time (CPU seconds)		Time reduction (%) vs. eC algorithm	Points where the algorithm reached <i>TILIM</i>		Points where both reached <i>TILIM</i>	Points where ReC-1B had better gap	Average Gap (%) for common cases, where the algorithm reached <i>TILIM</i>	
	eC	ReC-1B		ReC-1B	eC			ReC-1B	eC
1	75488.1	71868.7	4.8	17	16	14	12	7.17	5.71
2	36938.7	30803.5	16.6	4	3	3	3	4.94	4.10
3	55727.3	56616.8	-1.6	12	11	10	7	7.17	6.17
4	57796.9	57010.1	1.4	11	11	10	7	5.80	4.78
5	70297.0	69771.8	0.7	17	16	16	12	6.51	5.63
6	52441.5	52481.9	-0.1	10	12	10	10	6.79	4.45
7	40767.2	36204.2	11.2	8	7	6	4	5.10	3.60
8	45126.7	36820.1	18.4	6	4	3	2	6.25	3.10
9	26588.0	20870.6	21.5	4	1	1	1	7.00	0.63
10	49415.2	40121.6	18.8	10	7	7	5	5.79	5.46

The comparison between the eC and ReC-1B algorithms for this group of instances can be summarized as follows:

- About the run time:
 - In 8 out of 10 instances (80%) ReC-1B was faster than eC.
 - Only for the favorable cases this improvement can be up to 21.5% of time reduction and 11.7% in the average.
- About the quality of the solution:
 - For the iterations (ε^t values) where both algorithms reached the time limit, in 63 out of 80 points (78.8%) the ReC-1B algorithm found a better gap than eC.

Figure 5 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. It is evident the tradeoff between cost (f_1) and time (f_2).

Figure 6 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^t.

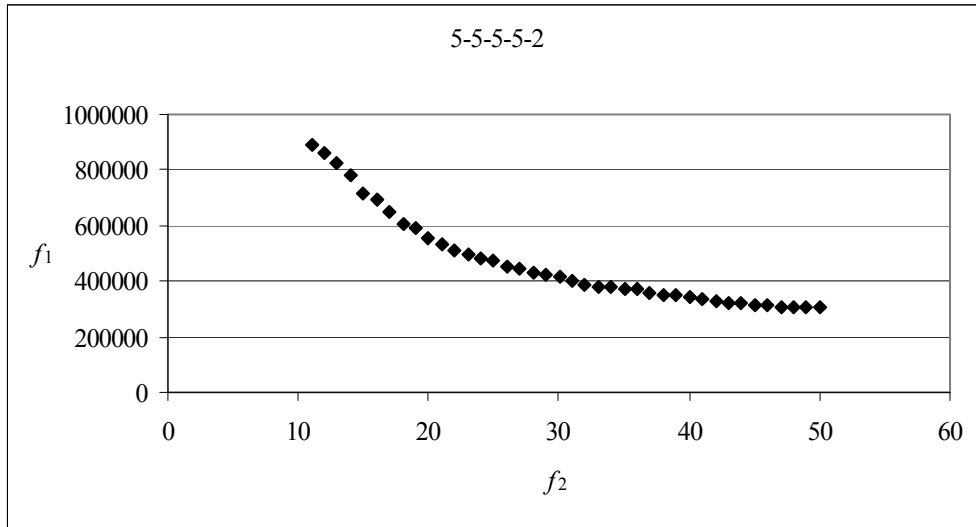


Figure 5. Set of non-dominated solutions for the instance 5-5-5-2.

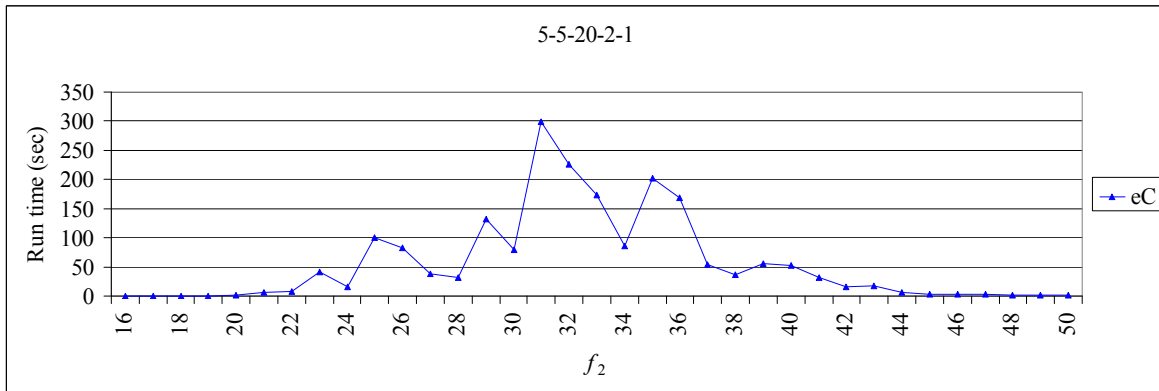


Figure 6. Run time behavior of eC algorithm as a function of the value of ϵ^t .

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 Section 6. In summary, scheme LP is the linear programming relaxation of the MIP. Scheme LPc is the LP relaxation with cuts. 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 and compiled with Visual Studio 6.0. CPLEX 9.1 callable library (ILOG SA, 2005) 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.

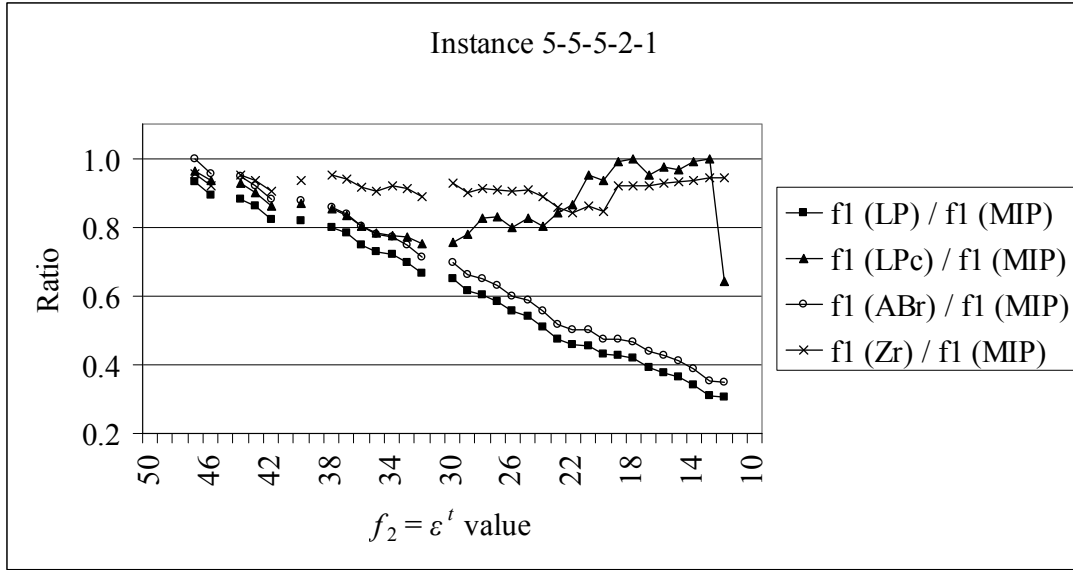


Figure 7. Quality of the lower bound sets for instance 5-5-5-2-1.

To define the quality of the lower bound set the schemes were tested in instances with known true efficient solution sets. Five instances of the sizes 5-5-5-2, 5-5-5-5, and 5-5-20-2 were studied. 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 Figures 7, 8 and 9 for instances 5-5-5-2-1, 5-5-5-5-1 and 5-5-20-2-1 respectively. This trend is similar for the other instances examined.

Table 4 presents the average, minimum and maximum values of the f_1 (scheme) / f_1 (MIP) ratios for the lower bound schemes proposed. 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 time for the linear relaxation schemes are displayed in Table 5. The column MIP represents the time required by the ReC-1B algorithm to obtain the optimal solution.

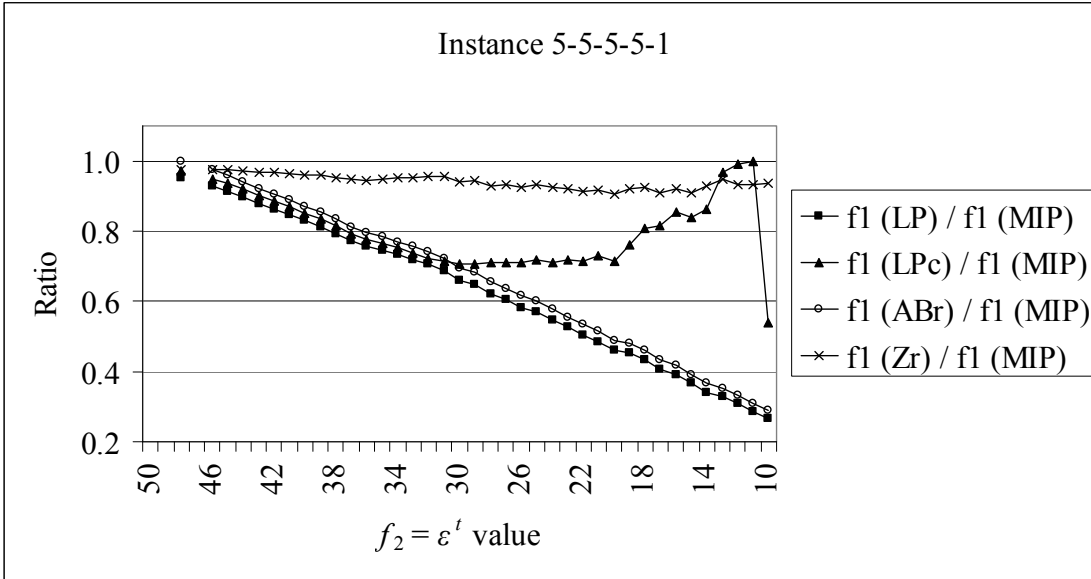


Figure 8. Quality of the lower bound sets for instance 5-5-5-5-1.

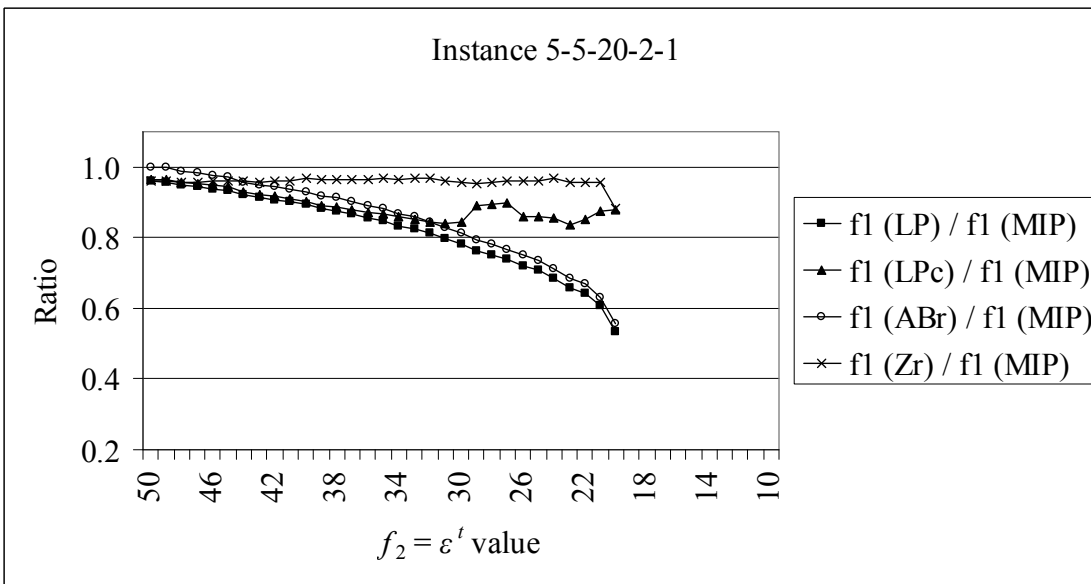


Figure 9. Quality of the lower bound sets for instance 5-5-20-2-1.

In the case of the LPc scheme a recovery can be observed in the lowest and highest values of ε^t . The reason may be the default use heuristics and cuts in CPLEX to improve the lower bound 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.

Table 4. 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	0.598	0.867	0.649	0.913	0.304	0.643	0.350	0.841	0.933	1.000	1.000	0.956
	2	0.684	0.841	0.761	0.916	0.375	0.739	0.418	0.871	0.897	0.970	0.998	0.963
	3	0.606	0.832	0.726	0.891	0.327	0.767	0.403	0.836	0.835	0.963	0.994	0.944
	4	0.629	0.808	0.707	0.859	0.331	0.679	0.386	0.808	0.892	1.000	1.000	0.905
	5	0.557	0.848	0.667	0.871	0.318	0.750	0.414	0.828	0.846	1.000	1.000	0.928
5-5-5-5	1	0.622	0.802	0.656	0.941	0.267	0.538	0.288	0.905	0.953	1.000	1.000	0.977
	2	0.679	0.825	0.719	0.950	0.324	0.713	0.347	0.911	0.946	1.000	1.000	0.971
	3	0.641	0.789	0.728	0.903	0.278	0.654	0.318	0.850	0.882	0.964	1.000	0.937
	4	0.637	0.781	0.678	0.915	0.288	0.661	0.310	0.879	0.941	0.992	1.000	0.951
	5	0.627	0.799	0.652	0.914	0.235	0.650	0.258	0.778	0.965	0.998	1.000	0.965
5-5-20-2	1	0.819	0.892	0.853	0.958	0.535	0.835	0.557	0.882	0.959	0.964	0.998	0.970
	2	0.801	0.880	0.828	0.959	0.533	0.811	0.552	0.923	0.965	0.971	0.999	0.976
	3	0.783	0.870	0.800	0.952	0.516	0.783	0.528	0.905	0.977	0.980	0.998	0.979
	4	0.784	0.854	0.809	0.958	0.507	0.762	0.526	0.926	0.967	0.969	0.998	0.978
	5	0.771	0.854	0.806	0.943	0.490	0.787	0.513	0.894	0.957	0.974	1.000	0.969

Table 5. Run time (CPU sec.) for the lower bounding schemes.

Lower bounding schemes						
Group code	Instance	MIP	LP	LPc	ABr	Zr
5-5-5-2	1	6.33	0.06	0.14	3.02	61.83
	2	11.25	0.05	0.12	2.61	53.36
	3	10.36	0.05	0.12	3.28	44.92
	4	17.66	0.05	0.18	4.69	98.28
	5	9.47	0.05	0.17	3.24	63.33
5-5-5-5	1	141.31	0.09	0.31	7.09	141840.50
	2	334.77	0.09	0.33	6.33	98279.72
	3	346.50	0.08	0.38	10.36	140449.33
	4	359.17	0.08	0.37	6.50	157355.89
	5	139.05	0.08	0.33	5.70	109087.41
5-5-20-2	1	1981.89	0.16	0.41	8.83	2711.25
	2	1658.08	0.14	0.41	11.72	7957.02
	3	12709.45	0.16	0.46	9.48	253041.64
	4	2436.19	0.17	0.46	11.41	35591.46
	5	3676.30	0.14	0.55	16.20	43516.73

8 Conclusions

In the process of supply chain design many decisions have to be made and several aspects must be taken into account. However, an area of opportunity was identified by incorporating the selection of the transportation channel in the distribution network design. Although several works have considered this decision before, few models studied the effect of this decision in the total cost and transportation time simultaneously (Arntzen et al., 1995; Zeng, 1998; Graves and Willems, 2005; Chang, Chung and Choy, 2006). Those few models that consider this simultaneous decision are limited because they do not generate a set of efficient solutions and simplify the problem treating one function as a constraint or combining both functions in an aggregated objective.

In our work we are addressing this issue by treating cost and time functions in a bi-objective programming framework. Additionally, these models do not include the facility location component. In this sense, the problem introduced in our paper considers the facility location and the transportation mode selection decisions, generating a set of efficient solutions for the cost-time tradeoff. We have named this new problem as “Capacitated Fixed Cost Facility Location Problem with Transportation Choices” (CFCLP-TC). In this problem the design of the supply chain is based on decisions about facilities to be opened, the flows between facilities and the selection of transportation channels between facilities. The problem is modeled as a bi-objective mixed-integer program. The model seeks to minimize both the total cost and the maximum delivery time from the plants to the customers. The total cost is a combination of transportation cost and fixed opening cost. 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, 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). In the eC algorithm, the value ε^t of the time function was reduced sequentially. In the ReC-2B and ReC-1B algorithms, the value of ε^t is increased. By proceeding this way in each optimization step, the previous solution can be used as incumbent solution, which is used advantageously within a branch-and-bound algorithm to solve the mixed-integer program involved. The difference between the ReC-2B and ReC-1B algorithms is in the way the initial and final values of ε^t are computed. In the ReC-2B algorithm, these values are obtained through independent optimization of each objective function. In the ReC-1B algorithm, the initial value of ε^t is estimated from a lower limit calculated from the instance data. 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 a start solution in each optimization step, in comparison to the eC algorithm. Also, the ReC-1B algorithm obtains the start value of ε^t from a lower limit estimated with the instance data, avoiding a time overhead that occurs in ReC-2B. Even for large instances the ReC-1B algorithm was able to obtain best gaps than the eC algorithm in most of the cases for the

hardest iterations. We infer that the ReC-1B algorithm may be adapted to other bi-objective problems easily. The exact algorithms were successfully applied for instances of up to 255 binary variables and 940 constraints. For larger instances, computer memory was insufficient to solve the MIP related to some values of ε^l .

In addition, four different strategies for obtaining lower bound sets were examined. These were based on linear relaxations of the MIP model. One scheme is the linear relaxation of the complete MIP model (LP). The second one is the linear relaxation with cuts added (LPc). The others make the linear relaxation of variables A_{ijl} and B_{jkl} (ABr), and of variables Z_j (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. One explanation for this is the difficulty CPLEX observes in identifying useful cuts in the relaxed problems, which may be caused by the poor numerical representation of the polyhedron. The other schemes produced low quality bound sets and should not be considered. Therefore 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.

Because of the computational complexity of the CFCLP-TC a heuristic method must be employed to solve larger instances. A modification of the ReC-1B algorithm was used to obtain reference sets (upper bound sets) for large instances. The heuristic rule was to impose a time limit in each main iteration (i.e, when fixing the value of ε^l .)

In summary, the first contribution of this work is the introduction of a model that considers the selection of the transportation channel in the supply chain design under a bi-objective optimization approach. This has never been addressed before to the best of our knowledge. We also provide a proof about the complexity of the problem. Another contribution is a detailed empirical study of three versions of the ε -constraint method for this bi-objective combinatorial optimization problem. One of these versions consistently found the optimal efficient frontiers for many of the instances tested. Our belief is that this method can be extended to other bi-objective combinatorial optimization problems with similar characteristics. We also proposed and empirically evaluated some lower bounding schemes. One of the procedures tested found lower bounds of better quality than those found by the LP relaxation. Further work for speeding up the solution time of this scheme is needed.

Acknowledgements: This research has been supported by ITESM Research Fund CAT128 and the Mexican National Council for Science and Technology (grant SEP-CONACYT 61903).

References

C. H. Aikens (1985). Facility location models for distribution planning. *European Journal of Operational Research*, 22(3):263-279.

- F. Altıparmak, M. Gen, L. Lin, and T. Paksoy (2006). A genetic algorithm approach for multi-objective optimization of supply chain networks. *Computers & Industrial Engineering*, 51(1):197-216.
- B. C. Arntzen, G. C. Brown, T. P. Harrison, and L. L. Trafton (1995). Global supply chain management at Digital Equipment Corporation. *Interfaces*, 25(1):69-93.
- R. H. Ballou (1999). *Business Logistics Management*. Prentice Hall, Upper Saddle River, USA.
- W. J. Baumol and P. Wolfe (1958). A warehouse-location problem. *Operations Research*, 6(2):252-263.
- J. Benjamin (1990). An analysis of mode choice for shippers in a constrained network with applications to just-in-time inventory. *Transportation Research Part B*, 24(3):229-245.
- F. T. S. Chan, S. H. Chung, and K. L. Choy (2006). Optimization of order fulfillment in distribution network problems. *Journal of Intelligent Manufacturing*, 17(3):307-319.
- S. Chopra and P. Meindl (2004). *Supply Chain Management: Strategy, Planning, and Operation*. Prentice Hall, Upper Saddle River, USA.
- J. F. Cordeau, F. Pasin, and M. M. Solomon (2006). An integrated model for logistics network design. *Annals of Operations Research*, 144(1):59-82.
- G. Cornuejols, G. L. Nemhauser, and L. A. Wolsey (1990). The uncapacitated facility location problem. In P. B. Mirchandani and R. L. Francis (editors), *Discrete Location Theory*, Chapter 3, pp. 119-171. Wiley, New York, USA.
- J. Current, H. Min, and D. Schilling (1990). Multiobjective analysis of facility location decisions. *European Journal of Operational Research*, 49(3):295-307.
- M. Ehrgott (2005). *Multicriteria Optimization*. Springer-Verlag, Berlin, Germany.
- M. Ehrgott and X. Gandibleux (2007). Bound sets for biobjective combinatorial optimization problems. *Computers & Operations Research*, 34(9):2674-2694.
- E. Eskigun, R. Uzsoy, P. V. Preckel, G. Beaujon, S. Krishnan, and J. D. Tew (2005). Outbound supply chain network design with mode selection, lead times and capacitated vehicle distribution centers. *European Journal of Operational Research*, 165(1):182-206.
- S. C. Graves and S. P. Willems (2005). Optimizing the supply chain configuration for new products. *Management Science*, 51(8):1165-1180.
- ILOG SA (2005). *ILOG CPLEX Callable Library 9.1 Reference Manual*. ILOG, France.

A. Klose and A. Drexl (2005). Facility location models for distribution system design. *European Journal of Operational Research*, 162(1):4-29.

A. A. Kuehn and M. J. Hamburger (1963). A heuristic program for locating warehouses. *Management Science*, 9(4):643-666.

D. Simchi-Levi, P. Kaminski, and E. Simchi-Levi (2000). *Designing and Managing the Supply Chain: Concepts, Strategies and Case Studies*. McGraw Hill, Boston, USA.

R. E. Steuer (1989). *Multiple Criteria Optimization: Theory, Computation and Application*. Krieger Publishing Company, Malabar, USA.

D. J. Thomas and P. M. Griffin (1996). Coordinated supply chain management. *European Journal of Operational Research*, 94(1):1-15.

T. H. Truong and F. Azadivar (2005). Optimal design methodologies for configuration of supply chains. *International Journal of Production Research*, 43(11):2217- 2236.

C. J. Vidal and M. Goetschalckx (1997). Strategic production-distribution models: A critical review with emphasis on global supply chain models. *European Journal of Operational Research*, 98(1):1-18.

W. Wilhelm, D. Liang, B. Rao, D. Warriar, X. Zhu, and S. Bulusu (2005). Design of international assembly systems and their supply chains under NAFTA. *Transportation Research Part E*, 41(6):467-493.

D. D. Zeng (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, December.

G. Zhou, H. Min, and M. Gen (2003). A genetic algorithm approach to the bi-criteria allocation of customers to warehouses. *International Journal of Production Economics*, 86(1):35-45.