An Integrated Model for Logistics Network Design

JEAN-FRANÇOIS CORDEAU and FEDERICO PASIN

HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal, Canada H3T 2A7

MARIUS M. SOLOMON

Northeastern University
360 Huntington Avenue
Boston, MA 02115 U.S.A.

August 2003

To appear in *Annals of Operations Research*

Abstract

In this paper we introduce a new formulation of the logistics network design problem encountered in deterministic, single-country, single-period contexts. Our formulation is flexible and integrates location and capacity choices for plants and warehouses with supplier and transportation mode selection, product range assignment and product flows. We next describe two approaches for solving the problem - a simplex-based branch-and-bound and a Benders decomposition approach. We then propose valid inequalities to strengthen the LP relaxation of the model and improve both algorithms. The computational experiments we conducted on realistic randomly generated data sets show that Benders decomposition is somewhat more advantageous on the more difficult problems. They also highlight the considerable performance improvement that the valid inequalities produce in both solution methods. Furthermore, when these constraints are incorporated in the Benders decomposition algorithm, this offers outstanding reoptimization capabilities.

Keywords: logistics, network design, Benders decomposition.

1 Introduction

In recent years, the constant emphasis on productivity gains and customer satisfaction has led to rapidly evolving business environments characterized by time compressed supply chains, alliances, and mergers and acquisitions. In turn, these have highlighted the importance of properly designing or redesigning the production and distribution networks of manufacturing firms. A growing emphasis on e-collaboration, technologically advanced manufacturing, and just-in-time pick-ups and deliveries is also amplifying the role of supply chain management as a strategic tool for competitiveness. As a result, a number of firms have relied on optimization techniques for decision support when planning their logistics activities (see, e.g., [4], [11], [17] and [18]).

This paper addresses the problem of designing the supply chain or logistics network of a manufacturing firm operating in a single-country environment. A logistics network is a set of suppliers, manufacturing plants and warehouses organized to manage the procurement of raw materials, their transformation into finished products, and the distribution of finished products to customers. Usually, the planning of a logistics network involves making decisions regarding:

- the number, location, capacity and technology of manufacturing plants and warehouses;
- the selection of suppliers;
- the assignment of product ranges to manufacturing plants and warehouses;
- the selection of distribution channels and transportation modes;
- the flows of raw materials, semi-finished and finished products through the network.

These decisions can be classified into three categories according to their importance and the length of the planning horizon considered. First, choices regarding the location, capacity and technology of plants and warehouses are generally seen as *strategic* with a planning horizon of several years. Second, supplier selection, product range assignment as well as distribution channel and transportation mode selection belong to the *tactical* level and can be revised every few months. Finally, raw material, semi-finished and finished product flows in the network are *operational* decisions that are easily modified in the short term.

The *logistics network design problem* (LNDP) consists of making the above-mentioned decisions so as to satisfy customer demands while minimizing the sum of fixed and variable costs associated with procurement, production, warehousing and transportation. Because of its complexity, it is often decomposed into several components treated separately. For instance, one may separate strategic, tactical and operational decisions or divide the network in several parts according to product categories or geographical considerations. However, given the importance of the interactions between these decisions, important benefits can be obtained by treating the network as a whole and considering its various components simultaneously.

Although there exists an abundant literature on capacitated facility location problems (see, e.g., [1], [7] and [13]), very few models address the LNDP in its entirety. Following the pioneering work of Geoffrion and Graves [8] on multi-commodity distribution network design, numerous models have been developed to locate facilities by taking into account several production, transportation and warehousing issues. An interesting example is the work of Pirkul and Jayaraman [16] on integrated production, transportation and distribution planning. However, as can be seen from the recent reviews by Geoffrion and Powers [9], Thomas and Griffin [19] and Vidal and Goetschalckx [20], most location models do not incorporate at least some aspects of the problem such as supplier or transportation mode selection.

One of the first efforts to integrate procurement, production and distribution decisions belongs to Cohen and Lee [5] who developed a detailed model for logistics network design in a global (i.e., international) context. The model considers a single planning period with deterministic demand and is solved by a hierarchical approach in which integer variables associated with the design of the network are first assigned values so as to obtain a simple linear program. A multi-period model for the LNDP in a global context was later proposed by Arntzen et al. [2]. Besides dealing with typical international issues such as local content and offset trade constraints, the model can handle an arbitrary number of production and distribution stages. A sophisticated solution methodology based on elastic constraints, row factorization, cascaded problem solution and constraint-branching enumeration was used to solve the model which has been applied at Digital Equipment Corporation. Very recently, Dogan and Goetschalckx [6] described a comprehensive multi-period model for the LNDP in a single-country environment. The model integrates strategic issues such as facility location and sizing with tactical decisions concerning production, inventory and customer allocation. It is solved by a Benders decomposition approach in which the subproblem separates into a set of network flow problems.

The contribution of this paper is to introduce a general and flexible formulation of the LNDP for the deterministic, single-country, single-period context, and describe two approaches for solving the problem: a simplex-based branch-and-bound approach and a Benders decomposition approach. Furthermore, we propose valid inequalities to strengthen the LP relaxation of the model and improve both algorithms. The formulation extends previous work by integrating location and capacity choices for plants and warehouses with supplier and transportation mode selection, product range assignment and product flows. Its structure makes it easy to impose several types of configuration constraints such as single-sourcing requirements. It can also be adapted to handle several problem extensions such as multiple planning periods or stochastic demand. While the formulation can be solved efficiently by using a commercial integer programming solver, it is also well suited for a primal decomposition approach such as Benders decomposition. The latter approach is particularly useful because of the reoptimization capabilities it provides when performing "what-if" analyses.

The rest of the paper is organized as follows. The next section presents a mathematical formulation of the LNDP and then Section 3 describes the solution methodology. Computational experiments are reported in Section 4, followed by our conclusions and extensions discussed in the final section.

2 Mathematical Formulation

Let $\mathcal{F}$ be the set of finished products. An element $f \in \mathcal{F}$ identifies either a specific article manufactured or assembled by the company, or a family of similar articles that can be aggregated and treated as a single product for planning purposes. Let $\mathcal{R}$ denote the set of raw materials and purchased components or supplies used in the manufacturing or assembly of finished products. For every $r \in \mathcal{R}$ and every $f \in \mathcal{F}$, let b^{rf} be the quantity of raw material r required in the production of one unit of product f. The set of all suppliers considered by the company is denoted by $\mathcal{S}$, and $\mathcal{S}^r \subseteq \mathcal{S}$ represents the subset of suppliers that are eligible to provide raw material $r \in \mathcal{R}$. Let also $\mathcal{P}$ and $\mathcal{W}$ denote the sets of actual and potential locations for plants and warehouses, respectively. For every product $f \in \mathcal{F}$, let $\mathcal{P}^f$ and $\mathcal{W}^f$ denote the subsets of plants and warehouses at which product f can be made and stored, respectively. Finally, let $\mathcal{C}$ be the set of customer locations. Again, an element $c \in \mathcal{C}$ may identify either a specific customer or a group of customers (i.e., a customer zone) that may be aggregated for planning purposes. For every $c \in \mathcal{C}$ and every $f \in \mathcal{F}$, let a_c^f be the demand of customer c for product f.

For notational convenience, denote by $\mathcal{K} = \mathcal{R} \cup \mathcal{F}$ the set of all commodities represented in the model, and by $\mathcal{O} = \mathcal{S} \cup \mathcal{P} \cup \mathcal{W}$ and $\mathcal{D} = \mathcal{P} \cup \mathcal{W} \cup \mathcal{C}$ the sets of origins and destinations for these commodities. Then, for every $k \in \mathcal{K}$, define $\mathcal{O}^k \subseteq \mathcal{O}$ and $\mathcal{D}^k \subseteq \mathcal{D}$ as the sets of potential origins and destinations for commodity k. More specifically, one has $\mathcal{O}^r = \mathcal{S}^r$ for any raw material $r \in \mathcal{R}$, and $\mathcal{O}^f = \mathcal{P}^f \cup \mathcal{W}^f$ for any product $f \in \mathcal{F}$. Similarly, possible destinations for a raw material r are plants at which products that require this raw material can be made, i.e., $\mathcal{D}^r = \cup_{f \in \mathcal{F}^r} \mathcal{P}^f$, where $\mathcal{F}^r = \{f \in \mathcal{F} | b^{rf} > 0\}$. Finally, the set of possible destinations for a product f is defined as $\mathcal{D}^f = \mathcal{W}^f \cup \mathcal{C}^f$ where $\mathcal{C}^f = \{c \in \mathcal{C} | a_c^f > 0\}$.

For every $k \in \mathcal{K}$ and every $o \in \mathcal{O}^k$, let V_o^k be a binary variable, with cost c_o^k, taking the value 1 if and only if commodity k is assigned to origin o. For instance, variable V_s^r would take the value 1 if supplier s is selected to provide raw material r, and variable V_p^f would take the value 1 if product f is made at plant p. For every origin $o \in \mathcal{O}$, also define a binary variable U_o equal to 1 if and only if this origin is assigned at least one commodity, and let c_o be the fixed cost of selecting this origin. In the case of a supplier $s \in S$, the variable U_s

would take the value 1 if the supplier is selected to provide at least one raw material. In the case of a potential plant or warehouse location, the associated variable would take the value 1 if the corresponding location is chosen to site a facility. For every $k \in \mathcal{K}$, $o \in \mathcal{O}^k$ and $d \in \mathcal{D}^k$, let Y_{od}^k be a binary variable, with cost c_{od}^k, equal to 1 if and only if origin o provides commodity k to destination d. For every $k \in \mathcal{K}$ and $o \in \mathcal{O}^k$, let q_o^k be an upper limit on the amount of commodity k to be provided by origin o to any destination and let q_{od}^k be the maximum to be provided to destination d. Finally, for every $o \in \mathcal{O}$, let u_o be the capacity, in equivalent units, of origin o, and for every $k \in \mathcal{K}$, let u_o^k be the amount of capacity required by one unit of commodity k at origin o. In the case of a plant p, u_p would represent the total manufacturing capacity in the planning period while u_p^f would be the transformation factor to convert real units of product f into equivalent units.

For every origin-destination pair $(o, d) \in \mathcal{O} \times \mathcal{D}$, let $\mathcal{M}_{od}$ be the set of transportation modes between o and d. Then, for every $m \in \mathcal{M}_{od}$, define a binary variable Z_{od}^m equal to 1 if and only if transportation mode m is used between origin o and destination d. Let c_{od}^m be the fixed cost of using mode m, and let g_{od}^m be its capacity. For every $k \in \mathcal{K}$, $o \in O^k$ and $d \in D^k$, let $\mathcal{M}_{od}^k \subseteq \mathcal{M}_{od}$ be the set of feasible transportation modes between o and d for commodity k, and let g^{km} be the capacity usage of one unit of commodity k in mode m. Then, for every $m \in \mathcal{M}_{od}^k$, define a non-negative variable X_{od}^{km}, with cost c_{od}^{km}, representing the number of units of commodity k transported from origin o to destination d using mode m. For instance, X_{pw}^{fm} is the amount of product f transported from plant p to warehouse w using mode $m \in \mathcal{M}_{pw}^f$. Because a single planning period is considered, the total amount of product p manufactured at plant p in this period is given by $\sum_{w \in \mathcal{W}} \sum_{m \in \mathcal{M}_{pw}} X_{pw}^{fm}$.

Let $\mathbb{B}$ denote the set of integers $\{0, 1\}$. The model can then be stated as follows:

Minimize

$$\sum_{o \in \mathcal{O}} \left[c_o U_o + \sum_{d \in \mathcal{D}} \sum_{m \in \mathcal{M}_{od}} c_{od}^m Z_{od}^m \right] + \sum_{k \in \mathcal{K}} \sum_{o \in \mathcal{O}^k} \left[c_o^k V_o^k + \sum_{d \in \mathcal{D}^k} \left[c_{od}^k Y_{od}^k + \sum_{m \in \mathcal{M}_{od}^k} c_{od}^{km} X_{od}^{km} \right] \right] \quad (1)$$

subject to

$$\sum_{s \in \mathcal{S}^r} \sum_{m \in \mathcal{M}_{sp}^r} X_{sp}^{rm} - \sum_{f \in \mathcal{F}^r} \sum_{w \in \mathcal{W}^f} \sum_{m \in \mathcal{M}_{pw}^f} b^{rf} X_{pw}^{fm} = 0 \qquad r \in \mathcal{R}; p \in \mathcal{P} \quad (2)$$

$$\sum_{p \in \mathcal{P}^f} \sum_{m \in \mathcal{M}_{pw}^f} X_{pw}^{fm} - \sum_{c \in \mathcal{C}^f} \sum_{m \in \mathcal{M}_{wc}^f} X_{wc}^{fm} = 0 \qquad f \in \mathcal{F}; w \in \mathcal{W}^f \quad (3)$$

$$\sum_{w \in \mathcal{W}^f} \sum_{m \in \mathcal{M}_{wc}^f} X_{wc}^{fm} = a_c^f \quad f \in \mathcal{F}; c \in \mathcal{C}^f \quad (4)$$

$$\sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{D}^k}\sum_{m\in\mathcal{M}_{od}^k} u_o^k X_{od}^{km} - u_o U_o \leq 0 \quad o \in \mathcal{O} \tag{5}$$

$$\sum_{d\in\mathcal{D}^k}\sum_{m\in\mathcal{M}_{od}^k} X_{od}^{km} - q_o^k V_o^k \leq 0 \quad k \in \mathcal{K}; o \in \mathcal{O}^k \tag{6}$$

$$\sum_{m\in\mathcal{M}_{od}^k} X_{od}^{km} - q_{od}^k Y_{od}^k \leq 0 \quad k \in \mathcal{K}; o \in \mathcal{O}^k; d \in \mathcal{D}^k \tag{7}$$

$$\sum_{k\in\mathcal{K}} g^{km} X_{od}^{km} - g_{od}^m Z_{od}^m \leq 0 \quad o \in \mathcal{O}; d \in \mathcal{D}; m \in \mathcal{M}_{od} \tag{8}$$

$$X_{od}^{km} \geq 0 \quad k \in \mathcal{K}; o \in \mathcal{O}^k; d \in \mathcal{D}^k; m \in \mathcal{M}_{od}^k \tag{9}$$

$$U_o \in \mathbb{B} \quad o \in \mathcal{O} \tag{10}$$

$$V_o^k \in \mathbb{B} \quad k \in \mathcal{K}; o \in \mathcal{O}^k \tag{11}$$

$$Y_{od}^k \in \mathbb{B} \quad k \in K; o \in \mathcal{O}^k; d \in \mathcal{D}^k \tag{12}$$

$$Z_{od}^m \in \mathbb{B} \quad o \in \mathcal{O}; d \in \mathcal{D}; m \in \mathcal{M}_{od}. \tag{13}$$

The objective function (1) minimizes the sum of all fixed and variable costs. Variable costs c_{od}^{km} may include not only transportation expenses but also relevant acquisition, production and storage costs. Constraints (2) ensure that the total amount of raw material r shipped to plant p is equal to the total amount required by all products made at this plant, while constraints (3) ensure that all finished products that enter a given warehouse also leave that warehouse. Demand constraints are imposed by equations (4). Constraints (5) impose global capacity limits on suppliers, plants and warehouses, whereas limits per commodity are enforced through (6). The latter constraints can be used to restrict the total amount of a given raw material that is purchased from a particular supplier or the number of units of a finished product that are made in a particular plant. Constraints (7) ensure that units of commodity k are not transported from o to d unless origin o is selected to provide the commodity to destination d. Finally, capacity constraints on individual transportation modes are imposed by (8).

Model (1)-(13) can be extended in several ways to handle various additional realistic situations. First, it is worth mentioning that by reversing the inequality sign, constraints similar to (5)-(8) can be used to impose lower limits on acquisition, production, storage and transportation activities. Such constraints can be used, for example, when a minimum amount of raw material must be purchased from a supplier to obtain a quantity discount. They can also be used to model situations where a minimum amount of finished product must be manufactured for a plant to be economically viable.

Second, if several capacity or technology choices are considered for a potential plant or warehouse location, these options can be modeled by defining several copies of the same location with different capacities, u_o and q_o^k, and different fixed and variable costs. A similar approach can be used to model quantity discounts offered by suppliers. It also applies to transportation modes which can be replicated to represent the same physical link with different capacities and costs.

Of course, if a given supplier, plant or warehouse must be selected, then the corresponding U_o variable can explicitly be set to 1 in the model. This is useful in the case of existing facilities which should remain active or when some location decisions are made with respect to criteria that are not taken into account by the model. The same reasoning also applies to transportation mode variables Z_{od}^m and assignment variables V_o^k and Y_{od}^k.

Additional network configuration constraints can also be introduced in the model. For example, if the total number of plants to be operated must lie between $\underline{n}$ and $\overline{n}$, these limits can be enforced with the simple constraint:

$$\underline{n} \leq \sum_{p \in \mathcal{P}} U_p \leq \overline{n}. \tag{14}$$

Similarly, if $\underline{n}^r$ and $\overline{n}^r$ are lower and upper limits on the total number of suppliers that should supply raw material $r \in \mathcal{R}$, then these limits can be imposed by the constraints:

$$\underline{n}^r \leq \sum_{o \in \mathcal{O}^r} V_o^r \leq \overline{n}^r \quad r \in \mathcal{R}. \tag{15}$$

Finally, single-sourcing for commodity k at destination d can be imposed with the constraint:

$$\sum_{o \in \mathcal{O}^k} Y_{od}^k \leq 1. \tag{16}$$

Single-sourcing constraints can be used, for example, to ensure that for each product $f \in \mathcal{F}$ and each customer $c \in \mathcal{C}^f$, the demand of the customer for the particular product is entirely satisfied from a unique warehouse.

Model (1)-(13) assumes a single manufacturing stage and a single distribution stage. These assumptions are easily relaxed by extending the network structure and modifying constraints (2) and (3) accordingly. In the case of seasonal demand, several planning periods can also be considered by defining X_{od}^{kmt} variables, where t denotes the period number, and introducing additional end-of-period inventory variables. Finally, the formulation can be adapted to handle stochastic demand in the form of an enumerable set of scenarios. These extensions will not be addressed in this paper but will be the object of subsequent research.

Table 1: Summary of notation

a_c^f	Demand of customer c for product f
b^{rf}	Amount of raw material r in product f
c_o	Fixed cost of selecting origin o
c_o^k	Fixed cost of assigning commodity k to origin o
c_{od}^k	Fixed cost of providing commodity k to destination d from origin o
c_{od}^m	Fixed cost of using transportation mode m between o and d
c_{od}^{km}	Unit cost for providing commodity k to d from o with mode m
g_{od}^m	Capacity of mode m between o and d
g^{km}	Amount of capacity required by one unit of commodity k in mode m
q_o^k	Upper limit on the amount of commodity k shipped from origin o
q_{od}^k	Upper limit on the amount of commodity k shipped from o to d
u_o	Capacity of origin o in equivalent units
u_o^k	Amount of capacity required by one unit of commodity k at origin o
$\mathcal{C}$	Set of customers
$\mathcal{C}^f$	Set of customers that require product f
$\mathcal{D}$	Set of destinations
$\mathcal{D}^k$	Set of potential destinations for commodity k
$\mathcal{F}$	Set of finished products
$\mathcal{F}^r$	Set of finished products that require raw material r
$\mathcal{K}$	Set of commodities
$\mathcal{M}_{od}$	Set of transportation modes between o and d
$\mathcal{M}_{od}^k$	Set of modes between o and d for commodity k
$\mathcal{O}$	Set of origins
$\mathcal{O}^k$	Set of potential origins for commodity k
$\mathcal{P}$	Set of potential plant locations
$\mathcal{P}^f$	Set of potential plant locations for making product f
$\mathcal{R}$	Set of raw materials
$\mathcal{S}$	Set of potential suppliers
$\mathcal{S}^r$	Set of potential suppliers providing raw material r
$\mathcal{W}$	Set of potential warehouse locations
$\mathcal{W}^f$	Set of potential warehouse locations for storing product f
X_{od}^{km}	Amount of commodity k provided by o to d with mode m
U_o	$= 1$ if origin o is selected
V_o^k	$= 1$ if commodity k is assigned to origin o
Y_{od}^k	$= 1$ if origin o provides commodity k to destination d
Z_{od}^m	$= 1$ if mode m is selected between o and d

3 Solution Methodology

Model (1)-(13) can be solved by a branch-and-bound approach in which lower bounds are computed by the simplex algorithm. However, its structure is also well suited for a primal decomposition approach such as Benders decomposition [3]. We first present this approach in Section 3.1, and then introduce valid inequalities that strengthen the LP relaxation and improve the performance of both solution approaches in Section 3.2.

3.1 Benders decomposition

For given values of the U, V, Y and Z variables that satisfy integrality constraints (10)-(13), model (1)-(13) reduces to the following *primal subproblem* involving only the X_{od}^{km} variables:

$$\text{Minimize} \quad \sum_{k\in\mathcal{K}} \sum_{o\in\mathcal{O}^k} \sum_{d\in\mathcal{D}^k} \sum_{m\in\mathcal{M}_{od}^k} c_{od}^{km} X_{od}^{km} \tag{17}$$

$$\sum_{s\in\mathcal{S}^r} \sum_{m\in\mathcal{M}_{sp}^r} X_{sp}^{rm} - \sum_{f\in\mathcal{F}^r} \sum_{w\in\mathcal{W}^f} \sum_{m\in\mathcal{M}_{pw}^f} b^{rf} X_{pw}^{fm} = 0 \qquad r\in\mathcal{R}; p\in\mathcal{P} \tag{18}$$

$$\sum_{p\in\mathcal{P}^f} \sum_{m\in\mathcal{M}_{pw}^f} X_{pw}^{fm} - \sum_{c\in\mathcal{C}} \sum_{m\in\mathcal{M}_{wc}^f} X_{wc}^{fm} = 0 \qquad f\in\mathcal{F}; w\in\mathcal{W}^f \tag{19}$$

$$\sum_{w\in\mathcal{W}^f} \sum_{m\in\mathcal{M}_{wc}^f} X_{wc}^{fm} = a_c^f \qquad f\in\mathcal{F}; c\in\mathcal{C}^f \tag{20}$$

$$\sum_{k\in\mathcal{K}} \sum_{d\in\mathcal{D}^k} \sum_{m\in\mathcal{M}_{od}^k} u_o^k X_{od}^{km} \le u_o \bar{U}_o \qquad o\in\mathcal{O} \tag{21}$$

$$\sum_{d\in\mathcal{D}^k} \sum_{m\in\mathcal{M}_{od}^k} X_{od}^{km} \le q_o^k \bar{V}_o^k \qquad k\in\mathcal{K}; o\in\mathcal{O}^k \tag{22}$$

$$\sum_{m\in\mathcal{M}_{od}^k} X_{od}^{km} \le q_{od}^k \bar{Y}_{od}^k \qquad k\in\mathcal{K}; o\in\mathcal{O}^k; d\in\mathcal{D}^k \tag{23}$$

$$\sum_{k\in\mathcal{K}} g^{km} X_{od}^{km} \le g_{od}^m \bar{Z}_{od}^m \qquad o\in\mathcal{O}; d\in\mathcal{D}; m\in\mathcal{M}_{od} \tag{24}$$

$$X_{od}^{km} \ge 0 \qquad k\in\mathcal{K}; o\in\mathcal{O}^k; d\in\mathcal{D}^k; m\in\mathcal{M}_{od}^k. \tag{25}$$

Let $\boldsymbol{\alpha} = (\alpha_p^r | r\in\mathcal{R}; p\in\mathcal{P})$, $\boldsymbol{\beta} = (\beta_w^f | f\in\mathcal{F}; w\in\mathcal{W})$, $\boldsymbol{\gamma} = (\gamma_c^f | f\in\mathcal{F}; c\in\mathcal{C})$, $\boldsymbol{\delta} = (\delta_o \le 0 | o\in\mathcal{O})$, $\boldsymbol{\zeta} = (\zeta_o^k \le 0 | k\in K; o\in\mathcal{O}^k)$, $\boldsymbol{\eta} = (\eta_{od}^k \le 0 | k\in K; o\in\mathcal{O}^k; d\in\mathcal{D}^k)$ and $\boldsymbol{\theta} = (\theta_{od}^m \le 0 | o\in\mathcal{O}; d\in\mathcal{D}; m\in\mathcal{M}_{od})$ be the dual variables associated with constraints (18)-(24), respectively.

The dual of the primal subproblem, called the *dual subproblem*, can be written as:

$$\text{Maximize} \quad \sum_{f\in\mathcal{F}}\sum_{c\in\mathcal{C}^f} a_c^f\gamma_c^f + \sum_{o\in\mathcal{O}}\left[u_o\bar{U}_o\delta_o + \sum_{d\in\mathcal{D}}\sum_{m\in\mathcal{M}_{od}} g_{od}^m\bar{Z}_{od}^m\theta_{od}^m\right] + \sum_{k\in\mathcal{K}}\sum_{o\in\mathcal{O}^k}\left[q_o^k\bar{V}_o^k\zeta_o^k + \sum_{d\in\mathcal{D}^k} q_{od}^k\bar{Y}_{od}^k\eta_{od}^k\right] \quad (26)$$

subject to

$$(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{\delta},\boldsymbol{\zeta},\boldsymbol{\eta},\boldsymbol{\theta}) \in \boldsymbol{\Delta}, \quad (27)$$

where $\boldsymbol{\Delta}$ denotes the polyhedron defined by the constraints of the problem.

The polyhedron $\boldsymbol{\Delta}$ does not depend on the values of the binary variables U, V, Y and Z which appear only in the objective function of the dual subproblem. Because all X_{od}^{km} variables are non-negative in the primal subproblem, the dual subproblem has one constraint of the form $\leq c_{od}^{km}$ for each variable X_{od}^m. If all cost coefficients c_{od}^{km} are non-negative, the dual subproblem is always feasible because the null vector $\mathbf{0}$ is a feasible solution. Hence, either the primal subproblem is infeasible or it is feasible and bounded. Let $P_{\boldsymbol{\Delta}}$ and $Q_{\boldsymbol{\Delta}}$ be the sets of real-valued vectors representing the extreme points and extreme rays of $\boldsymbol{\Delta}$, respectively.

For given values of the U, V, Y and Z variables, the dual subproblem is bounded and the primal subproblem is feasible if

$$\sum_{f\in\mathcal{F}}\sum_{c\in\mathcal{C}^f} a_c^f\gamma_c^f + \sum_{o\in\mathcal{O}}\left[u_o\bar{U}_o\delta_o + \sum_{d\in\mathcal{D}}\sum_{m\in\mathcal{M}_{od}} g_{od}^m\bar{Z}_{od}^m\theta_{od}^m\right] + \sum_{k\in\mathcal{K}}\sum_{o\in\mathcal{O}^k}\left[q_o^k\bar{V}_o^k\zeta_o^k + \sum_{d\in\mathcal{D}^k} q_{od}^k\bar{Y}_{od}^k\eta_{od}^k\right] \leq 0 \quad (28)$$

for all extreme rays $(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{\delta},\boldsymbol{\zeta},\boldsymbol{\eta},\boldsymbol{\theta}) \in Q_{\boldsymbol{\Delta}}$. In this case, the optimal value of both problems is given by the expression

$$\max_{(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{\delta},\boldsymbol{\zeta},\boldsymbol{\eta},\boldsymbol{\theta})\in P_{\boldsymbol{\Delta}}} \sum_{f\in\mathcal{F}}\sum_{c\in\mathcal{C}^f} a_c^f\gamma_c^f + \sum_{o\in\mathcal{O}}\left[u_o\bar{U}_o\delta_o + \sum_{d\in\mathcal{D}}\sum_{m\in\mathcal{M}_{od}} g_{od}^m\bar{Z}_{od}^m\theta_{od}^m\right] + \sum_{k\in\mathcal{K}}\sum_{o\in\mathcal{O}^k}\left[q_o^k\bar{V}_o^k\zeta_o^k + \sum_{d\in\mathcal{D}^k} q_{od}^k\bar{Y}_{od}^k\eta_{od}^k\right] \quad (29)$$

which is the maximum, over all extreme points of $\boldsymbol{\Delta}$, of the dual subproblem objective function (26).

Let $\boldsymbol{\mathcal{MP}}$ represent the set of configuration and integrality constraints on U, V, Y and Z variables. This set can contain any constraints, such as those of the form (14)-(16), that involve only the binary variables. Introducing the free variable λ, one thus obtains the following Benders master problem:

$$\text{Minimize} \quad \sum_{o\in\mathcal{O}}\left[c_oU_o + \sum_{d\in\mathcal{D}}\sum_{m\in\mathcal{M}_{od}} c_{od}^m Z_{od}^m\right] + \sum_{k\in\mathcal{K}}\sum_{o\in\mathcal{O}^k}\left[c_o^kV_o^k + \sum_{d\in\mathcal{D}^k} c_{od}^k Y_{od}^k\right] + \lambda \tag{30}$$

subject to

$$\sum_{f\in\mathcal{F}}\sum_{c\in\mathcal{C}^f} a_c^f\gamma_c^f + \sum_{o\in\mathcal{O}}\left[u_o\delta_oU_o + \sum_{d\in\mathcal{D}}\sum_{m\in\mathcal{M}_{od}} g_{od}^m\theta_{od}^m Z_{od}^m\right] +$$
$$\sum_{k\in\mathcal{K}}\sum_{o\in\mathcal{O}^k}\left[q_o^k\zeta_o^kV_o^k + \sum_{d\in\mathcal{D}^k} q_{od}^k\eta_{od}^k Y_{od}^k\right] \leq 0 \qquad (\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{\delta},\boldsymbol{\zeta},\boldsymbol{\eta},\boldsymbol{\theta}) \in Q_{\boldsymbol{\Delta}} \tag{31}$$

$$\lambda \geq \sum_{f\in\mathcal{F}}\sum_{c\in\mathcal{C}^f} a_c^f\gamma_c^f + \sum_{o\in\mathcal{O}}\left[u_o\delta_oU_o + \sum_{d\in\mathcal{D}}\sum_{m\in\mathcal{M}_{od}} g_{od}^m\theta_{od}^m Z_{od}^m\right] +$$
$$\sum_{k\in\mathcal{K}}\sum_{o\in\mathcal{O}^k}\left[q_o^k\zeta_o^kV_o^k + \sum_{d\in\mathcal{D}^k} q_{od}^k\eta_{od}^k Y_{od}^k\right] \qquad (\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{\delta},\boldsymbol{\zeta},\boldsymbol{\eta},\boldsymbol{\theta}) \in P_{\boldsymbol{\Delta}} \tag{32}$$

$$(U,V,Y,Z) \in \boldsymbol{\mathcal{MP}}. \tag{33}$$

Formulation (30)-(33) contains a very large number of constraints. However, an efficient solution method is obtained by dynamically generating only subsets of *feasibility cuts* (31) and *optimality cuts* (32). Starting from empty subsets of extreme points and extreme rays, each iteration of the algorithm first solves a *relaxed Benders master problem* which consists of model (30)-(33), where the sets $P_{\boldsymbol{\Delta}}$ and $Q_{\boldsymbol{\Delta}}$ are replaced by the subsets $P_{\boldsymbol{\Delta}}^\tau \subseteq P_{\boldsymbol{\Delta}}$ and $Q_{\boldsymbol{\Delta}}^\tau \subseteq Q_{\boldsymbol{\Delta}}$ of extreme points and extreme rays available at iteration τ. Solving the relaxed Benders master problem provides a lower bound LB on the optimal solution value as well as a solution $(\bar{U},\bar{V},\bar{Y},\bar{Z})$ which is used to set up the dual subproblem (26)-(27). If the dual subproblem is bounded, an optimal solution corresponding to an extreme point of $\boldsymbol{\Delta}$ can be identified and leads to an optimality cut of the form (32). In this case, an upper bound UB on the optimal solution value can be computed, and a feasible solution to the original problem can be identified by solving the primal subproblem (17)-(25). If the dual subproblem is unbounded, an extreme ray that violates one of the constraints (31) can be identified. After adding the newly identified extreme point or extreme ray to the appropriate set, the algorithm moves to iteration $\tau + 1$. The process continues until $LB = UB$, at which point an optimal solution has been identified. More details on this approach can be found in the original paper of Benders [3] and in application papers such as those of Geoffrion and Graves [8] and Dogan and Goetschalckx [6].

3.1.1 Generating Pareto-optimal cuts

When the primal subproblem (17)-(25) is degenerate, the dual subproblem (26)-(27) may have several optimal solutions, possibly yielding different optimality cuts of the form (32). Let $\boldsymbol{\phi} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}, \boldsymbol{\zeta}, \boldsymbol{\eta}, \boldsymbol{\theta})$ denote an extreme point of the set $P_{\boldsymbol{\Delta}}$. Let also $rhs(\boldsymbol{\phi})$ denote the right-hand-side of (32) for the extreme point $\boldsymbol{\phi}$. The cut obtained from the extreme point $\boldsymbol{\phi}^1$ dominates that obtained from the extreme point $\boldsymbol{\phi}^2$ if, for every $(U, V, Y, Z) \in \boldsymbol{\mathcal{MP}}$, $rhs(\boldsymbol{\phi}^1) \geq rhs(\boldsymbol{\phi}^2)$, with strict inequality for at least one point in $\boldsymbol{\mathcal{MP}}$. A cut is said to be Pareto-optimal if no other cut dominates it (see, e.g., [14]).

Let $\boldsymbol{\mathcal{MP}}^{LP}$ denote the polyhedron obtained by replacing the set $\mathbb{B}$ by the interval $[0,1]$ in (10)-(13), and let $ri(\boldsymbol{\mathcal{MP}}^{LP})$ denote the relative interior of $\boldsymbol{\mathcal{MP}}^{LP}$.

For a given vector $(\bar{U}, \bar{V}, \bar{Y}, \bar{Z}) \in \boldsymbol{\mathcal{MP}}^{LP}$ for which the dual subproblem is bounded, let $v(\bar{U}, \bar{V}, \bar{Y}, \bar{Z})$ denote the optimal value of the subproblem. To identify an optimal solution to the dual subproblem that yields a Pareto-optimal cut, one must solve the following auxiliary subproblem, where $(\tilde{U}, \tilde{V}, \tilde{Y}, \tilde{Z}) \in ri(\boldsymbol{\mathcal{MP}}^{LP})$:

$$\text{Maximize} \quad \sum_{f \in \mathcal{F}} \sum_{c \in \mathcal{C}^f} a_c^f \gamma_c^f + \sum_{o \in \mathcal{O}} \left[u_o \tilde{U}_o \delta_o + \sum_{d \in \mathcal{D}} \sum_{m \in \mathcal{M}_{od}} g_{od}^m \tilde{Z}_{od}^m \theta_{od}^m \right] + \sum_{k \in \mathcal{K}} \sum_{o \in \mathcal{O}^k} \left[q_o^k \tilde{V}_o^k \zeta_o^k + \sum_{d \in \mathcal{D}^k} q_{od}^k \tilde{Y}_{od}^k \eta_{od}^k \right] \tag{34}$$

subject to

$$\sum_{f \in \mathcal{F}} \sum_{c \in \mathcal{C}^f} a_c^f \gamma_c^f + \sum_{o \in \mathcal{O}} \left[u_o \bar{U}_o \delta_o + \sum_{d \in \mathcal{D}} \sum_{m \in \mathcal{M}_{od}} g_{od}^m \bar{Z}_{od}^m \theta_{od}^m \right] + \sum_{k \in \mathcal{K}} \sum_{o \in \mathcal{O}^k} \left[q_o^k \bar{V}_o^k \zeta_o^k + \sum_{d \in \mathcal{D}^k} q_{od}^k \bar{Y}_{od}^k \eta_{od}^k \right] = v(\bar{V}, \bar{Y}, \bar{Z}, \bar{U}) \tag{35}$$

$$(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}, \boldsymbol{\zeta}, \boldsymbol{\eta}, \boldsymbol{\theta}) \in \boldsymbol{\Delta}. \tag{36}$$

The additional constraint (35) ensures that one will choose an extreme point from the set of optimal solutions to the original dual subproblem. Let q be the dual variable associated with constraint (35). Instead of solving model (34)-(36), one can solve its dual which is easily obtained by introducing the extra variable q in model (17)-(25) and modifying its right-hand-side. Solving the auxiliary problem in this form is very convenient in terms of ease of implementation and computational efficiency since the same basic representation can be used to solve both the primal subproblem (17)-(25) and the auxiliary subproblem that is used to generate Pareto-optimal cuts.

3.1.2 Generating a set of initial cuts from problem relaxations and computing integer solutions

Instead of solving the integer relaxed master problem at every iteration of the Benders decomposition algorithm, one may first solve the LP relaxation of the problem by relaxing the integrality constraints on the master problem variables (see, e.g., [15]). Once the LP relaxation is solved, integrality constraints are reintroduced and additional cuts are generated until an optimal integer solution is found. The cuts generated when solving the LP relaxation are valid for the integer programming problem because the relaxation of integrality constraints on master problem variables has no effect on the subproblem.

The same idea can be used when configuration constraints are imposed on the binary variables. For example, if single-sourcing constraints (16) are imposed, these constraints can first be relaxed so as to generate optimality and feasibility cuts by solving a smaller, relaxed Benders master problem. Once an optimal solution has been reached for this relaxation, the single-sourcing constraints are reintroduced and more cuts are generated until an optimal solution is found.

Finally, to accelerate the solution of the integer master problem, branching priorities can be used so as to first make branching decisions on U_o variables followed by V_o^k, Y_{od}^k and Z_{od}^m variables, in that order.

3.2 Valid Inequalities

When solving model (1)-(13) either with a simplex-based branch-and-bound algorithm or with the Benders decomposition approach outlined in Section 3.1, various types of valid inequalities can be added to the formulation. For both approaches, these constraints can strengthen the LP relaxation of the problem. In the case of the Benders decomposition approach, they can also improve convergence by helping the relaxed master problem to find solutions that are close to optimal. Indeed, because the iterative algorithm is initialized from empty subsets of extreme rays and extreme points, the relaxed master problem initially contains only the integrality constraints. As a result, several iterations must be performed before enough information is transferred to the master problem. Introducing valid inequalities in the master problem can thus dramatically reduce the number of cuts that will have to be generated from extreme points and extreme rays of the dual subproblem polyhedron.

To strengthen the LP relaxation of model (1)-(13), the following constraints can be added to the formulation:

$$V_o^k \leq U_o \quad (k \in \mathcal{K}; o \in \mathcal{O}^k). \tag{37}$$

Constraints (37) ensure that a commodity k is not assigned to a source $o \in \mathcal{O}^k$ unless the source is also selected. Assuming that u_o is finite and u_o^k is positive for every k, constraints

(37) are redundant in the presence of (5). However, they may considerably strengthen the LP relaxation when u_o is large compared to the amount of capacity that is actually used in the solution. Observe that in presence of constraints (37), integrality constraints on the U_o variables can in fact be relaxed.

Recalling that a_c^f denotes the demand of customer c for product f, one may also add the constraints

$$\sum_{s\in\mathcal{S}^r} q_s^r V_s^r \geq \sum_{f\in F}\sum_{c\in\mathcal{C}^f} a_c^f b^{rf} \quad (r\in\mathcal{R}) \tag{38}$$

$$\sum_{p\in\mathcal{P}^f} q_p^f V_p^f \geq \sum_{c\in\mathcal{C}^f} a_c^f \quad (f\in\mathcal{F}) \tag{39}$$

$$\sum_{w\in\mathcal{W}^f} q_w^f V_w^f \geq \sum_{c\in\mathcal{C}^f} a_c^f \quad (f\in\mathcal{F}) \tag{40}$$

to ensure that enough capacity per raw material or per finished product is provided by the selected suppliers, plants and warehouses to satisfy the demand for all products. In addition, if the same system of equivalent units is used throughout the logistics network, the following constraints can be added to ensure that enough global capacity is provided by the selected suppliers, plants and warehouses:

$$\sum_{s\in\mathcal{S}} u_s U_s \geq \sum_{r\in\mathcal{R}} u^r \sum_{f\in\mathcal{F}} b^{rf} \sum_{c\in\mathcal{C}} a_c^f \tag{41}$$

$$\sum_{p\in\mathcal{P}} u_p U_p \geq \sum_{f\in\mathcal{F}} u^f \sum_{c\in\mathcal{C}} a_c^f \tag{42}$$

$$\sum_{w\in\mathcal{W}} u_w U_w \geq \sum_{f\in\mathcal{F}} u^f \sum_{c\in\mathcal{C}} a_c^f. \tag{43}$$

The latter two sets of constraints do not strengthen the LP relaxation of the problem. However, they considerably improve convergence when using Benders decomposition. In addition, their introduction results in less nodes being explored when using the simplex-based branch-and-bound approach.

When single-sourcing is imposed, the following constraints can be used to help ensure that the total demand of all customers assigned to a given warehouse does not exceed its capacity:

$$\sum_{c\in C} Y_{wc}^f a_c^f \leq q_w^f V_w^f \quad (w\in\mathcal{W}; f\in\mathcal{F}) \tag{44}$$

$$\sum_{c\in\mathcal{C}}\sum_{f\in\mathcal{F}} u_w^f a_c^f Y_{wc}^f \leq u_w U_w \quad (w\in\mathcal{W}). \tag{45}$$

Finally, when fixed costs and capacities g_{od}^m are imposed on transportation modes, the constraints

$$\sum_{w \in \mathcal{W}^f} Y_{wc}^f \geq 1 \qquad (f \in \mathcal{F}; c \in \mathcal{C}^f) \tag{46}$$

$$\sum_{p \in \mathcal{P}^f} Y_{pw}^f \geq V_w^f \quad (f \in \mathcal{F}; w \in \mathcal{W}^f) \tag{47}$$

$$\sum_{s \in \mathcal{S}^r} Y_{sp}^r \geq V_p^f \quad (f \in \mathcal{F}; p \in \mathcal{P}^f; r \in \mathcal{R}^f) \tag{48}$$

$$\sum_{m \in \mathcal{M}_{od}^k} Z_{od}^m \geq Y_{od}^k \quad (k \in \mathcal{K}; o \in \mathcal{O}^k; d \in \mathcal{D}^k) \tag{49}$$

$$Y_{od}^k \leq V_o^k \quad (k \in \mathcal{K}; o \in \mathcal{O}^k; d \in \mathcal{D}^k) \tag{50}$$

can be added to the formulation to ensure that whenever a commodity k must be transported between an origin o and a destination d, at least one transportation mode in M_{od}^k is selected. Constraints (46)-(48) ensure that one source is selected for each customer demand, for each product assigned to a warehouse and for each raw material required to make a product that is assigned to a plant. Constraints (49) force the selection of at least one transportation mode for each source that is chosen. Finally, constraints (50) ensure that an origin o is not selected as a source for commodity k unless the commodity is actually assigned to that origin. These constraints strengthen the LP relaxation and have proven to be quite effective in computational testing.

4 Computational Experiments

To evaluate the tractability of model (1)-(13) and compare the performance of the two solution approaches proposed in Section 3, we performed computational experiments on a set of randomly generated test instances. The procedure used to generate these instances is first described in Section 4.1, followed by a summary of computational results in Section 4.2 and a discussion of reoptimization capabilities in Section 4.3.

4.1 Description of Data

We randomly generated a set of 24 instances according to assumptions that strike a balance between realism and ease of generation and reproducibility. Instances vary according to three main dimensions: size, complexity and cost structure. The size of an instance is given by the number of suppliers ($|\mathcal{S}|$), the number of potential plant locations ($|\mathcal{P}|$), the number of potential warehouse locations ($|\mathcal{W}|$), the number of customers ($|\mathcal{C}|$), the number of raw materials ($|\mathcal{R}|$), and the number of finished products ($|\mathcal{F}|$). For an instance with $|\mathcal{C}| = n$, we have set $|\mathcal{S}| = |\mathcal{P}| = |\mathcal{W}| = n/10$ and $|\mathcal{R}| = |\mathcal{F}| = n/5$. Three basic sizes were used in our experiments: $n = 100$, 200 and 300.

The complexity of an instance is itself determined by two factors: capacity structure and flow magnitude. The capacity structure is determined by the number of suppliers that can provide each raw material ($|\mathcal{S}^r|$), the number of potential plants that can make each product ($|\mathcal{P}^f|$) and the number of warehouses that can distribute each product ($|\mathcal{W}^f|$). For low capacity instances (denoted by the suffix 'c'), these values are chosen randomly in the set $\{1, \ldots, 5\}$ according to a uniform distribution, while for high capacity instances (denoted by the suffix 'C'), they are chosen in the set $\{1, \ldots, 10\}$. The corresponding number of items (suppliers, plants or warehouses) are then selected randomly (without replacement) according to a uniform distribution over the set of compatible items. For example, if $|\mathcal{S}^r| = 4$ for raw material r, then four suppliers will be selected at random from $\mathcal{S}$ to obtain $\mathcal{S}^r$.

For both low and high capacity instances, the actual overall and commodity specific capacities are determined as follows. For each commodity $k \in \mathcal{K}$, a unit capacity usage u^k is first generated by choosing a random integer from the set $\{1, \ldots, 10\}$ according to a uniform distribution. For every origin $o \in \mathcal{O}^k$, we assume $u_o^k = u^k$. Let u be the total manufacturing capacity that is required to satisfy the demand for all products and let $\bar{u} = u/|\mathcal{P}|$. The capacity u_p of each plant $p \in \mathcal{P}$ is selected at random from the set $[\alpha \cdot \bar{u}, \beta \cdot \bar{u}]$ according to a uniform distribution. For all instances, we have set $\alpha = 1$ and $\beta = |\mathcal{P}|$. The same approach is used to generate u_o values for the suppliers and warehouses. A similar method is also used to generate the u_o^k values that represent individual capacities for raw materials and finished products. In this case however, the average value $\bar{u}^k$ is computed with respect to the number of locations that can provide this commodity (i.e., $|\mathcal{O}|^k$). For low capacity instances, these rules tend to ensure that approximately 50% of all potential locations are selected and that each raw material and finished product is assigned to approximately 50% of the origins that can provide it. These percentages are closer to 25% for high capacity instances.

The flow magnitude is determined by the number of raw materials that go into each finished product ($|\mathcal{R}^f|$) and the number of customers that have a positive demand for each product ($|\mathcal{C}^f|$). For low flow magnitude instances (denoted by the suffix 'f'), the values of $|\mathcal{R}^f|$ and $|\mathcal{C}^f|$ are chosen from the sets $\{1, 5\}$ and $\{1, 25\}$, respectively, while for high flow magnitude instances (denoted by the suffix 'F'), these values are chosen from the sets $\{1, 10\}$ and $\{1, 50\}$, respectively. In both cases, the actual values a_c^f are chosen randomly from the set $\{1, \ldots, 10\}$, for every finished product $f \in \mathcal{F}$ and every customer $c \in \mathcal{C}^f$. In all instances, the amount b^{rf} of raw material $r \in \mathcal{R}^f$ that goes into each unit of finished product f is also chosen randomly from the set $\{1, \ldots, 10\}$.

The cost structure is determined as follows. For each plant $p \in \mathcal{P}$, a fixed cost c_p is first chosen randomly in the interval $[10^5, 10^6]$ according to a uniform distribution. Next, for each product $f \in \mathcal{F}$, an average fixed cost $\bar{c}^f$ is chosen randomly in the interval $[10^4, 10^5]$. Then, for every plant $p \in \mathcal{P}^f$, a fixed cost c_p^f is chosen from the set $[\alpha \cdot \bar{c}^f, \beta \cdot \bar{c}^f]$, where $\alpha = 0.75$ and $\beta = 1.25$. This ensures that the fixed cost of making product f varies from plant to plant within reasonable limits. For each warehouse $w \in \mathcal{W}$, fixed costs c_w and c_w^f are generated by using the same procedure and choosing values in $[10^4, 10^5]$ and $[10^3, 10^4]$, respectively. In the case of suppliers, the corresponding intervals are $[10^3, 10^4]$ and $[10^2, 10^3]$.

For every variable X_{od}^{km}, the variable cost c_{od}^{km} is composed of two distinct terms: the unit transportation cost of commodity k from o to d with mode m and the unit purchase, production or warehousing cost of commodity k at the origin o. For every commodity k, every origin $o \in \mathcal{O}^k$ and every destination $d \in \mathcal{D}^k$, an average unit transportation cost $\bar{t}_{od}^k$ is first generated by multiplying the Euclidean distance between o and d by a random number chosen according to a uniform distribution in the interval $[1, 10]$. For every location, Euclidean coordinates are themselves chosen randomly in the unit square $[0, 1] \times [0, 1]$. Then, for every mode $m \in \mathcal{M}_{od}^{km}$, a cost t_{od}^{km} is chosen from the interval $[\alpha \cdot \bar{t}_{od}^k, \beta \cdot \bar{t}_{od}^k]$, where $\alpha = 0.75$ and $\beta = 1.25$. Next, for every raw material $r \in \mathcal{R}$ and every finished product $f \in \mathcal{F}$, an average unit purchase, production or warehousing cost $\bar{a}^k$ is chosen randomly in the interval $[1, 10]$. Then, for every origin $o \in \mathcal{O}^k$, a unit cost a_o^k is chosen in the interval $[\alpha \cdot \bar{a}^k, \beta \cdot \bar{a}^k]$ where $\alpha = 0.75$ and $\beta = 1.25$. Finally, the cost c_{od}^{km} is obtained by setting $c_{od}^{km} = t_{od}^{km} + a_o^k$. For each size and complexity variant, we consider two levels of variable costs. For low variable cost instances (denoted by the suffix 'v'), variable costs are determined as explained above while for high variable cost instances (denoted by the suffix 'V'), these values are multiplied by 10. These rules ensure that variable costs represent 5-10% of total cost in the former case and 25-50% in the latter.

Finally, in all instances, a single transportation mode with no fixed cost is used between suppliers and plants as well as between plants and warehouses. However, for every warehouse-customer pair, the number of available transportation modes is selected randomly from the set $\{1, \ldots, 3\}$. These assumptions represent a situation where the company uses a single transportation mode (e.g., full truckload transportation) for all movements between plants and warehouses, but has a choice of transportation modes (with different fixed and variable costs) for the different customer zones it is serving. For each mode, a fixed cost c_{od}^m is then chosen randomly from the interval $[10^3, 10^4]$. For each finished product $f \in \mathcal{F}$, the value g^{fm} is set equal to 1. Then, the capacity g_{wc}^m of mode m is equal to the total demand (in real units) of customer c. As a result, the capacity constraints are not binding but their right-hand-sides serve as "big M" constants to impose the fixed cost c_{od}^m whenever a mode is used.

The three different sizes, two capacity structures and two demand structures yield a total of 12 basic instances for which two cost structures are considered. Table 4.1 summarizes the main characteristics and size of model (1)-(13) for each of these basic instances. The largest instance, 300CF, has a total of 19,232 binary variables, 23,725 continuous variables and

22,323 constraints. Because fixed costs are imposed only on transportation modes between warehouses and customers, Z_{od}^m variables are defined only for $(o, d) \in \mathcal{W} \times \mathcal{C}$. Furthermore, Y_{od}^k variables do not carry a fixed cost but are defined for the purpose of imposing single-sourcing constraints and introducing valid inequalities (46)-(50). It is worth mentioning that when fixed costs are not considered for transportation modes and single-sourcing is not imposed, the resulting model is considerably smaller because all Z_{od}^m and Y_{od}^k variables can be dropped from the formulation. For each instance, the number of constraints reported in the table does not include the sets of valid inequalities whose cardinality will be given separately in the next section.

Table 2: Characteristics and size of basic problem instances

					Number of variables				**Number of**
No.	$\lvert\mathcal{C}\rvert$	$\lvert\mathcal{R}\rvert$, $\lvert\mathcal{F}\rvert$	$\lvert\mathcal{S}\rvert, \lvert\mathcal{P}\rvert, \lvert\mathcal{W}\rvert$	U_o	V_o^k	Y_{od}^k	Z_{od}^m	X_{od}^{km}	**constraints**
100cf	100	20	10	30	162	935	1,309	2,346	2,905
100cF	100	20	10	30	171	1,296	1,334	3,212	3,525
100Cf	100	20	10	30	341	1,761	1,775	5,110	4,490
100CF	100	20	10	30	269	2,096	1,662	5,523	4,856
200cf	200	40	20	60	369	1,803	2,962	4,995	6,218
200cF	200	40	20	60	350	3,085	4,016	7,612	9,072
200Cf	200	40	20	60	567	2,851	4,164	8,556	8,859
200CF	200	40	20	60	606	4,778	5,403	13,580	12,657
300cf	300	60	30	90	539	2,558	4,465	7,072	9,194
300cF	300	60	30	90	520	4,386	6,823	11,567	14,404
300Cf	300	60	30	90	1009	4,825	7,636	16,243	15,560
300CF	300	60	30	90	942	7,920	10,280	23,725	22,323

The size of these instances is similar to or larger than the size of real-life instances solved in various applications in the literature. For example, Pooley [17] reports results for a network with 10 plant and 13 warehouse locations, 48 customer zones and 6 product types. Arntzen et al. [2] describe an application at Digital Equipment Corporation with 33 plant and 30 warehouse locations, leading to a model with approximately 6,000 constraints and 20,000 variables. Pirkul and Jayaraman [16] present results on randomly generated instances with up to 10 plant and 20 warehouse locations, 100 customer zones and 3 products. They also present results on real-life instances with 5 plant and 30 warehouse locations, 75 customer zones and 10 products. Finally, Camm et al. [4] report on a study at Procter & Gamble involving hundreds of suppliers, over 50 product lines, 60 plants, 10 distribution centers and hundreds of customer zones.

4.2 Summary of Results

For each of the 24 instances, we consider three scenarios: in the first, we do not impose either single-sourcing or fixed costs on transportation modes. In the second, we only require single-sourcing for each customer demand. Finally, the third supposes single-sourcing as well as fixed costs on all transportation modes between warehouses and customers. The first scenario is thus a relaxation of the second which, in turn, is a relaxation of the third.

All tests were performed on a Pentium III (933 MHz) processor with 256 Mb of RAM. For the simplex-based branch-and-bound approach, we used CPLEX 6.6.1 with steepest-edge pricing, strong branching and a depth-first search until an integer solution is found, followed by a best-bound search. These settings provided the best results throughout our experiments. For the Benders decomposition solution, CPLEX was used for solving the LP relaxations and the MIP problems. The same parameter settings as above were used for the simplex pricing and the branch-and-bound search.

When solving the problem with CPLEX, the branch-and-bound search was stopped when an integer solution within 1% of optimality was identified. Although it would be possible to solve the problem to optimality, computation times tend to grow considerably compared to those required to obtain near-optimal solutions. Given that the data (cost, demand and capacity estimations) used in real-life applications often contain a margin of error larger than 1%, we feel that solving the problem to optimality is rarely justified in practice.

For Benders decomposition, a two-phase approach was used as previously explained in Section 3.1.2. In the first phase, integrality was relaxed for the master problem variables and cuts were generated until $(UB - LB)/LB < 0.001$ (see Section 3.1). This is equivalent to solving the LP relaxation with a 0.1% optimality tolerance. In the second phase, integrality was imposed on the master problem variables, and the algorithm iteratively solved the integer master problem and generated additional cuts until an integer solution within 1% of optimality was identified. Generally, each second phase iteration takes much longer than a first phase iteration because the relaxed Benders master problem must be solved with integrality constraints in the former case. From the computational tests, we have observed that solving the LP relaxation with a larger optimality tolerance resulted in more cuts being generated in the second phase whereas decreasing the tolerance below 0.1% did not further reduce the number of iterations performed in that phase.

Finally, the Pareto-optimal cuts we generated for all instances and scenarios provided significant performance improvements over the standard implementation. However, each iteration took longer because the auxiliary subproblem had to be solved whenever the primal subproblem was feasible. Nevertheless, the total number of iterations performed was greatly reduced. On most instances, we observed tenfold speed improvements. Figure 1 shows the amelioration of the lower and upper bounds as a function of CPU time when Pareto-optimal cuts were used compared to when they were not, for a typical instance of the problem.

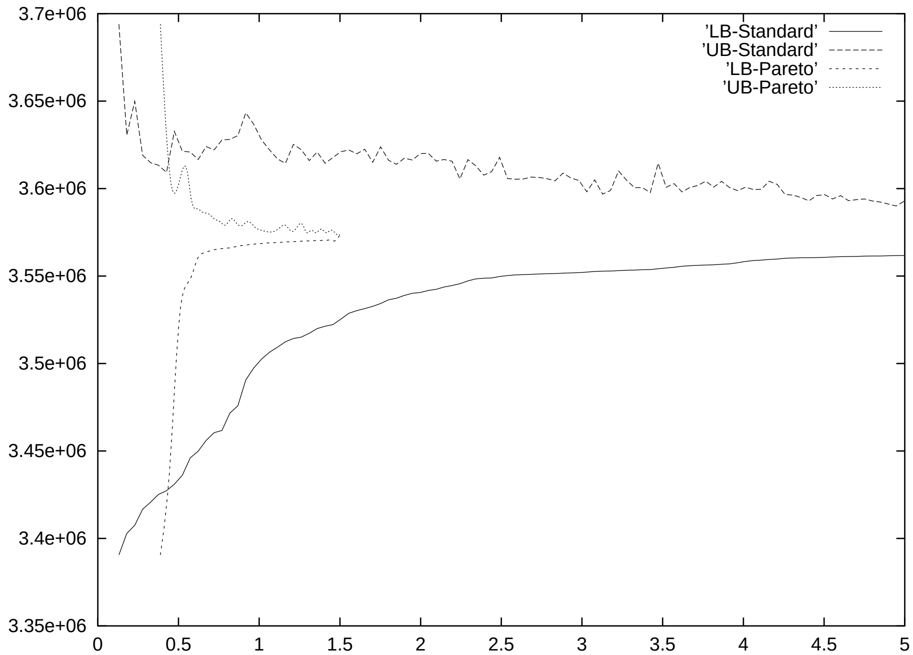

Figure 1: Values of lower (LB) and upper (UB) bounds as a function of CPU time

4.2.1 First scenario

Because this scenario relaxes single-sourcing constraints and fixed costs on transportation modes, variables Y_{od}^{k} and Z_{od}^{m} as well as constraints (7) and (8) are not required and can be omitted from the model.

As a first step in our experiments, we wanted to evaluate the impact of the valid inequalities (37) and (38)-(43) on solution time. For the CPLEX branch-and-bound approach, this is shown in Table 3. For the smallest eight instances, the columns under the heading *Basic model* report the CPU time (in minutes) needed to identify an integer solution within 1% of optimality, the number of branch-and-bound nodes explored and the (approximate) integrality gap for the model (1)-(13). The next two groups of columns report similar statistics when the constraints (37) are included either by themselves or together with (38)-(43). Column # indicates the total number of valid inequalities added to the model. The gaps reported may slightly overestimate the true integrality gaps because the search is stopped as soon as an integer solution within 1% of optimality is identified.

The results show that in most cases constraints (37) strengthened the LP relaxation and considerably reduced the difficulty of the problem. Constraints (38)-(43) also positively affected performance, dramatically reducing the number of branch-and-bound nodes that needed to be explored, even though they did not further strengthen the LP relaxation. The Benders decomposition could not solve even the smallest instances within 24 hours of CPU time without introducing both types of valid inequalities. Consequently, these two sets were used in all further testing.

Table 3: Impact of valid inequalities

	Basic model			**With (37)**				**With (37) and (38)-(43)**			
No.	**CPU**	**Nodes**	**Gap**	**CPU**	**Nodes**	**Gap**	#	**CPU**	**Nodes**	**Gap**	#
100cfv	4.40	2,794	66.61	1.18	732	47.51	162	0.03	13	47.36	225
100cfV	3.66	1,365	50.52	1.43	816	35.73	162	0.04	21	35.64	225
100cFv	34.44	16,653	69.08	84.57	38,801	50.11	171	0.04	12	50.03	234
100cFV	11.29	3,374	50.16	35.93	16,536	36.36	171	0.06	21	36.19	234
100Cfv	435.56	127,591	48.56	149.47	37,693	33.56	341	0.14	21	33.45	404
100CfV	177.71	51,864	36.50	21.95	5,239	25.21	341	0.15	21	25.32	404
100CFv	>720	>2E5		>720	>2E5		269	0.57	111	40.26	332
100CFV	174.03	48,574	32.06	18.50	5,213	22.46	269	0.46	91	22.74	332

Table 4 reports the results obtained by the Benders decomposition and CPLEX methods for all instances. For the former approach, columns *LP* and *MIP* indicate the number of cuts generated for the LP relaxation and the additional number of cuts generated for the mixed-integer problem. Columns *Feas.* and *Opt.* show the total number of feasibility and optimality cuts that were generated in the two phases. Column *LP* provides the CPU time (in minutes) required to solve the LP relaxation within 0.1% of optimality while column MIP gives the total CPU time required to find an integer solution within 1% of optimality. Since the value of the LP relaxation is the same in both approaches, we only report the (approximate) integrality gaps for the CPLEX. Of course, the cost of the solutions identified by the two solution methods (and the resulting integrality gaps) may differ by at most 1% because of the heuristic stopping criterion.

The results show that the performance of the two approaches is somewhat comparable. The average total CPU time is 0.86 minutes for CPLEX and 1.64 minutes for Benders decomposition. Interestingly, the latter approach is affected by the magnitude of the variable costs as a percentage of the total costs. When subproblem costs are larger, more information must be transferred to the master problem in the form of Benders cuts. This phenomenon is reflected by larger computation times and a larger number of optimality cuts for the 'V' problems when compared to their 'v' counterparts.

It is apparent from these results that the Benders decomposition method benefits from generating an initial set of cuts by solving the LP relaxation. Although the integrality gaps are rather large, only a few iterations need to be performed in the second phase of the

Table 4: Computational statistics for the first scenario

	Benders Decomposition						**CPLEX**			
		Benders Cuts			**CPU Time**		**CPU Time**			
No.	**LP**	**MIP**	**Feas.**	**Opt.**	**LP**	**MIP**	**LP**	**MIP**	**Nodes**	**Gap**
100cfv	34	2	32	4	0.02	0.03	0.01	0.03	13	47.36
100cfV	52	4	34	22	0.04	0.05	0.01	0.05	21	35.64
100cFv	32	1	28	5	0.03	0.04	0.01	0.04	12	50.03
100cFV	44	1	28	17	0.05	0.06	0.01	0.06	21	36.19
100Cfv	85	3	77	11	0.26	0.33	0.03	0.14	21	33.45
100CfV	110	3	84	29	0.36	0.43	0.03	0.15	21	25.32
100CFv	116	2	96	22	0.41	0.45	0.03	0.57	111	40.26
100CFV	159	10	117	52	0.60	0.76	0.03	0.46	91	22.74
200cfv	110	4	109	5	0.28	0.32	0.05	0.19	21	42.23
200cfV	120	8	108	20	0.37	0.47	0.04	0.30	48	34.46
200cFv	48	1	42	7	0.22	0.24	0.07	0.31	36	33.14
200cFV	78	1	60	19	0.36	0.39	0.07	0.38	41	21.11
200Cfv	87	1	83	5	0.39	0.42	0.11	0.59	60	42.81
200CfV	136	1	100	37	0.86	0.89	0.11	0.33	21	32.02
200CFv	185	1	168	18	1.97	2.06	0.17	1.00	58	38.93
200CFV	248	2	194	56	2.89	3.04	0.17	0.99	63	20.69
300cfv	64	2	62	4	0.25	0.29	0.10	0.49	50	37.90
300cfV	88	2	63	27	0.51	0.57	0.09	0.30	21	30.82
300cFv	83	2	76	9	0.66	0.73	0.16	0.79	53	33.75
300cFV	116	2	85	33	1.20	1.34	0.15	0.83	61	19.36
300Cfv	282	1	267	16	3.83	4.20	0.41	3.47	101	43.86
300CfV	308	1	227	82	6.08	6.35	0.41	2.38	101	33.03
300CFv	114	1	88	27	3.35	3.80	0.57	3.33	87	36.74
300CFV	286	4	96	194	11.16	12.04	0.52	3.41	121	17.92

algorithm when the Benders master problem must be solved as an integer program. This is explained by the fact that the cuts generated in the first phase provide a good approximation of the feasible region of the integer master problem.

4.2.2 Second scenario

In this scenario, Y_{ok}^{k} variables are added to the formulation together with constraints (7) and (16) to impose the single-sourcing of every customer demand.

Again, we first evaluated the impact of introducing additional valid inequalities. Table 5 compares the results obtained by the simplex-based branch-and-bound approach with and without constraints (44)-(45). Recall that in both cases, constraints (37)-(43) were added to the formulation. Here too, the introduction of a small number of valid inequalities had a major impact on performance. On the larger instances, both the CPU time and the number of nodes explored were reduced on average by a factor of 10. These constraints similarly influenced the Benders decomposition.

Table 5: Impact of additional valid inequalities for single-sourcing

	Basic model			**With (44)-(45)**			
No.	**CPU**	**Nodes**	**Gap**	**CPU**	**Nodes**	**Gap**	**#**
100cfv	1.08	342	47.36	0.44	99	47.36	70
100cfV	0.97	311	35.64	0.15	31	36.08	70
100cFv	1.86	440	50.03	0.44	74	50.03	63
100cFV	2.05	481	36.19	0.48	84	36.19	63
100Cfv	5.46	918	33.46	0.71	67	33.46	122
100CfV	10.69	1862	24.98	0.87	95	24.95	122
100CFv	22.92	3489	40.28	2.50	265	40.45	89
100CFV	15.80	2474	22.45	2.35	241	22.51	89

For Benders decomposition, single-sourcing constraints (16) affect only the master problem. As explained in Section 3.1.2, instead of introducing these constraints at the beginning of the solution process, one can first solve a relaxation of the problem obtained by introducing variables Y_{od}^k in the model but dropping constraints (16). All cuts generated when solving this relaxation are valid for the restricted problem because the presence of constraints (16) does not affect the polyhedron of the dual subproblem. In our tests, very few iterations (i.e., often less than 5) were needed to find a solution to the restricted problem after having solved this relaxation. As before, integrality constraints on the master problem are added last and a few additional iterations must be performed to obtain a near-optimal integer solution.

Table 6 presents the results obtained by both approaches for this scenario. For the Benders decomposition, we separately report the number of cuts generated for solving the initial relaxation (LP relaxation without single-sourcing constraints), followed by the number of additional cuts needed to solve the LP relaxation of the restricted problem, and the number of further cuts required to identify an integer solution within 1% of optimality. Except for three cases (200CFv, 200CFV and 300CFV), the total CPU time to find an integer solution within 1% of optimality was always smaller for the Benders decomposition. In addition, its average CPU time was 4.08 minutes compared to 11.66 minutes for the CPLEX. Of course, this difference is in part explained by the exceptionnally large CPU time for instance 300Cfv.

Table 6: Computational statistics for the second scenario

	Benders Decomposition							**CPLEX**			
		Benders Cuts				**CPU Time**		**CPU Time**			
No.	**Rel.**	**LP**	**MIP**	**Feas.**	**Opt.**	**LP**	**MIP**	**LP**	**MIP**	**Nodes**	**Gap**
100cfv	33	1	1	30	5	0.04	0.05	0.03	0.44	99	47.36
100cfV	52	1	1	31	23	0.08	0.11	0.03	0.15	31	36.08
100cFv	28	8	1	31	6	0.13	0.15	0.05	0.44	74	50.03
100cFV	36	1	1	20	18	0.10	0.11	0.05	0.48	84	36.19
100Cfv	105	2	2	96	13	0.61	0.71	0.12	0.71	67	33.46
100CfV	127	1	3	101	30	0.66	0.81	0.11	0.87	95	24.95
100CFv	110	1	1	89	23	0.66	0.80	0.15	2.50	265	40.45
100CFV	173	4	4	128	53	1.32	2.21	0.14	2.35	241	22.51
200cfv	104	2	1	101	6	0.42	0.49	0.16	2.01	217	42.26
200cfV	144	2	1	125	22	0.79	0.91	0.14	5.87	661	34.49
200cFv	52	2	1	46	9	0.59	0.72	0.31	3.45	265	33.14
200cFV	66	1	1	48	20	0.61	0.80	0.29	1.77	101	20.73
200Cfv	117	3	1	114	7	1.02	1.10	0.40	2.98	186	42.81
200CfV	142	2	1	106	39	1.57	1.65	0.42	3.20	193	31.98
200CFv	221	1	1	204	19	3.82	4.20	0.82	3.03	101	38.93
200CFV	260	3	2	206	59	5.64	6.05	0.82	2.94	101	20.68
300cfv	61	1	2	59	5	0.42	0.54	0.30	3.54	266	37.91
300cfV	93	1	2	68	28	0.80	0.94	0.29	15.23	1281	30.80
300cFv	99	1	2	92	10	1.38	1.76	0.68	5.80	252	33.56
300cFV	109	1	2	78	34	2.03	2.72	0.67	9.03	414	19.82
300Cfv	239	2	1	225	17	4.84	5.66	1.37	151.02	6275	43.83
300CfV	292	1	1	203	91	8.44	8.97	1.24	28.97	1153	32.94
300CFv	122	4	1	96	31	8.60	10.16	2.61	15.33	306	36.96
300CFV	296	10	10	105	211	25.72	46.29	2.44	17.74	388	17.78

4.2.3 Third scenario

In this last scenario, fixed costs and capacities are imposed on all transportation modes between warehouses and customers in addition to the previous single-sourcing requirement. As a result, mode selection variables Z_{od}^{km} must be introduced in the formulation together with capacity constraints (8).

As expected from the first two scenarios, valid inequalities proved to be extremely useful in improving the performance of both solution approaches. Since transportation modes must be chosen only between warehouses and customers, constraints (47)-(48) can be disregarded in these experiments. Furthermore, single-sourcing implies that constraints (46) are automatically satisfied in the presence of (16). Finally, constraints (49) are redundant when the valid constraints (44)-(45) are considered, but they do, however, strengthen the LP relaxation. As a result, our analysis of valid inequalities (46)-(50) concentrated on the latter two sets.

In this scenario, solving the problem without any of the additional constraints required several hours of computation, even for the smallest of the 24 instances. The addition of valid inequalities was thus absolutely necessary to obtain good quality solutions for the larger instances. Table 7 presents the results obtained with the additional constraints (49), and with both (49) and (50). Constraints (49) had a considerable effect, bringing CPU times down from several hours to a few minutes. The additional constraints (50) had a limited (and sometimes even negative) impact on small problems but did prove to be useful on the larger ones. They also strengthened the LP relaxation as shown by the reduced integrality gaps obtained. Finally, observe that there is exactly one constraint of each type for each variable Y_{od}^{k}. The main drawback of these constraints is thus their large number. For the Benders decomposition, we experimented with a dynamic generation of these constraints when they became violated. This did not lead to any improvement as more than 50% of all constraints were generated in the first few iterations when the optimal solution to the master problem tended to vary significantly from one iteration to the next.

We have also considered a successively restrictive Benders decomposition approach, where one starts by solving the relaxation obtained by dropping single-sourcing constraints and setting the fixed cost c_{od}^{m} of all transportation modes equal to 0. One then proceeds by solving each of the more restrictive problems obtained by sequentially reintroducing these constraint types and finally the integrality constraints on the master problem variables. Unfortunately, this did not prove advantageous. Because valid inequalities (49)-(50) restrict the problem and tighten the LP relaxation, we observed that far fewer iterations were performed when the single-sourcing and transportation mode fixed cost constraints were included right from

Table 7: Impact of additional valid inequalities for mode selection

	With (49)				**With (49)-(50)**			
No.	**CPU**	**Nodes**	**Gap**	**#**	**CPU**	**Nodes**	**Gap**	**#**
100cfv	0.63	32	43.10	935	0.62	39	38.87	1870
100cfV	0.79	45	33.62	935	0.68	47	30.25	1870
100cFv	1.07	40	45.48	1296	1.15	41	41.68	2592
100cFV	1.42	73	34.37	1296	1.06	48	31.21	2592
100Cfv	7.39	84	32.27	1761	6.46	74	23.97	3522
100CfV	15.41	248	24.93	1761	16.16	267	19.29	3522
100CFv	15.13	289	35.98	2096	31.24	620	31.76	4192
100CFV	16.66	337	21.92	2096	13.63	211	19.35	4192

the start. Even though each iteration took longer, the total CPU times was slightly reduced.

Table 8 shows comparative statistics for the two approaches. Again, Benders decomposition was on average faster than the simplex-based branch-and-bound method (22.69 minutes compared to 28.89 minutes). In all but one case (300CfV), the CPU time to find an integer solution within 1% of optimality was also smaller for the former approach than for the latter. As explained above, the reduced number of iterations compared to the previous two scenarios is a direct result of the presence of valid inequalities (49)-(50). Because these constraints strengthen the LP relaxation, integrality gaps are also smaller in this scenario relative to the other two. For this scenario, CPU times are sometimes very large. However, given the complexity of the problem and the size of the instances we considered, we believe that an investment of a few hours of computation time for a strategic planning problem is worthwhile and reasonable. This is particularly true since our approach lends itself to fast reoptimization following small changes in the data.

4.3 Reoptimization Capabilities

Since the LNDP is a strategic planning problem, for a solution methodology to be viable, it is utterly important that it be capable of efficient reoptimization in order to perform "what-if" analyses. Indeed, most planners generally examine several scenarios, such as comparing different demand and cost scenarios or different types of production and distribution network structures.

After first solving the problem with current demand levels, one might for example fix the values of the U_o variables and reoptimize the problem assuming a 10% increase in demand. Solving the problem again with the increased demand but leaving the U_o variables free would then provide an estimate of how far the best solution for the current demand is from optimality, if demand were to increase by 10%. The two reoptimizations can be efficiently solved by Benders decomposition since the two changes involved (fixing binary variables U_o and modifying constants a_c^f) do not affect the dual subproblem polyhedron. Indeed, fixing binary variables to 1 affects only the master problem while increasing demand affects only the objective function of the dual subproblem. As a result, all extreme points and extreme rays identified when first solving the problem are still valid and can be used to generate an initial set of optimality and feasibility cuts for the solution process. For a simplex-based branch-and-bound approach, however, the search for integer solutions must restart from the first node of the tree because the changes made affect the bounds that are computed at each node. Obviously, the basis of the LP optimal solution for the original problem can often be used as a starting point. However, our computational experiments showed that very little time is actually spent solving the LP relaxation.

Reoptimization capabilities are in fact extremely useful in a wide array of situations.

Table 8: Computational statistics for the third scenario

	Benders Decomposition						**CPLEX**			
	Benders Cuts				**CPU Time**		**CPU Time**			
No.	**LP**	**MIP**	**Feas.**	**Opt.**	**LP**	**MIP**	**LP**	**MIP**	**Nodes**	**Gap**
100cfv	23	1	20	4	0.10	0.15	0.20	0.62	39	38.87
100cfV	44	1	24	21	0.21	0.48	0.22	0.68	47	30.25
100cFv	17	1	13	5	0.15	0.25	0.45	1.15	41	41.68
100cFV	19	1	6	14	0.23	0.39	0.34	1.06	48	31.21
100Cfv	84	2	77	9	1.52	2.96	1.33	6.46	74	23.97
100CfV	87	1	57	31	1.60	3.42	1.44	16.16	267	19.29
100CFv	81	1	61	21	1.65	4.05	2.71	31.24	620	31.76
100CFV	100	2	63	39	2.43	8.13	2.56	13.63	211	19.35
200cfv	81	1	77	5	1.41	1.82	1.04	2.23	61	35.97
200cfV	73	1	52	22	1.34	2.00	0.99	2.47	82	29.80
200cFv	40	1	34	7	1.80	2.49	2.45	4.86	72	27.54
200cFV	54	1	34	21	2.23	4.03	2.52	5.82	101	18.13
200Cfv	76	1	73	4	2.84	5.11	4.83	11.40	101	33.91
200CfV	111	2	84	29	4.62	8.32	4.73	13.10	171	26.23
200CFv	186	1	172	15	18.14	27.61	9.32	48.64	321	26.64
200CFV	290	1	193	98	33.47	47.53	9.78	57.56	484	15.93
300cfv	46	2	44	4	1.55	1.92	1.80	3.49	61	29.96
300cfV	69	2	50	21	2.36	2.82	1.83	4.32	101	24.79
300cFv	87	2	81	8	5.18	6.94	4.72	7.87	71	25.36
300cFV	92	2	76	18	6.35	9.04	4.96	9.33	101	15.82
300Cfv	227	1	218	10	29.84	60.68	23.81	74.82	277	33.00
300CfV	273	1	202	72	49.80	114.93	19.86	89.40	460	25.82
300CFv	120	1	97	24	48.41	109.32	57.07	147.24	273	24.73
300CFV	136	1	101	36	56.62	120.21	56.80	140.01	358	13.86

Other common examples are the addition of configuration constraints such as a minimum number of plants to operate or a particular location that must be chosen to site a facility. Reoptimization is also interesting in contexts where the user wants to impose some decisions and let the solver optimize the rest of the network. With Benders decomposition, different partial configurations can be tested rapidly by reoptimization. The only changes that may require complete optimization from scratch are those that affect the cost of the flow variables X_{od}^{km} or the coefficients of these variables in the capacity constraints. These two types of changes affect the constraints of the dual subproblem and, as a result, the set of extreme points and extreme rays of the associated polyhedron. Other changes such as the modification of fixed costs associated with binary variables and the modification of capacity levels (u_o, u_o^k, g_{od}^m, ...) can be handled through reoptimization.

The results presented for the second scenario have already illustrated the reoptimization capabilities provided by Benders decomposition. Additional testing we performed with slight variations of the problem have further indicated that the problem can often be reoptimized in just a fraction of the total CPU time required to solve it from scratch.

5 Conclusions and Extensions

This paper has introduced a new integrated formulation for the logistics network design problem and compared two solution methodologies for it - a classical simplex-based branch-and-bound and a Benders decomposition approach. Our computational experiments showed that the methods are competitive and that Benders decomposition is slightly more advantageous on the more difficult problems. We also proposed several groups of valid inequalities and highlighted the considerable performance improvement they produce in both solution methods. Furthermore, when these constraints are incorporated in the Benders decomposition algorithm, this offers outstanding reoptimization capabilities.

We believe our results are general in nature and will remain valid independent of the scenario chosen. The experiments we have performed show that the methodology can be used to solve realistic instances of large size. Furthermore, the reasonable computation times and the good reoptimization capabilities of Benders decomposition lead us to believe that the proposed approach is applicable in contexts where solutions must be obtained quickly. Our methodology thus represents a likely alternative to meta-heuristics such as tabu search and simulated annealing that have also proven to be quite effective in terms of computation time but usually do not provide a precise measure of deviation from optimality (see, e.g., Lapierre et al. [12] and Jayaraman and Ross [10]).

The formulation presented here is flexible and can easily be adapted to handle multiple production and distribution stages as well as multiple technology and capacity alternatives at any given location. Future research could concentrate on extending the model and solution method to handle the cases of dynamic (time-varying) and stochastic demand. The first extension can be handled by discretizing the planning period and introducing additional inventory variables in the formulation. If these linking variables are retained in the Benders master problem, the subproblem decomposes by subperiod. The second extension can be handled as a stochastic program with recourse in which a small set of scenarios (e.g., pessimistic, realistic and optimistic) is considered. Benders decomposition should again be an appropriate method for the solution of such problems.

References

[1] C.H. Aikens. Facility location models for distribution planning. *European Journal of Operational Research*, 22:263–279, 1985.

[2] B.C. Arntzen, G.G. Brown, T.P. Harrison, and L.L. Trafton. Global supply chain management at Digital Equipment Corporation. *Interfaces*, 25(1):69–93, 1995.

[3] J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4:238–252, 1962.

[4] J.D. Camm, T.E. Chorman, F.A. Dill, J.R. Evans, D.J. Sweeney, and G.W. Wegryn. Blending OR/MS, judgment, and GIS: Restructuring P&G's supply chain. *Interfaces*, 27(1):128–142, 1997.

[5] M.A. Cohen and H.L. Lee. Resource deployment analysis of global manufacturing and distribution networks. *Journal of Manufacturing and Operations Management*, 2:81–104, 1989.

[6] K. Dogan and M. Goetschalckx. A primal decomposition method for the integrated design of multi-period production-distribution systems. *IIE Transactions*, 31:1027–1036, 1999.

[7] Z. Drezner, editor. *Facility Location*. Springer-Verlag, New York, 1995.

[8] A.M. Geoffrion and G.W. Graves. Multicommodity distribution system design by Benders decomposition. *Management Science*, 20:822–844, 1974.

[9] A.M. Geoffrion and R.F. Powers. Twenty years of strategic distribution system design: An evolutionary perspective. *Interfaces*, 25(5):105–127, 1995.

[10] V. Jayaraman and A. Ross. A simulated annearling methodology to distribution network design and management. *European Journal of Operational Research*, 144:629–645, 2003.

[11] M. Köksalan and H. Süral. Efes beverage group makes location and distribution decisions for its malt plants. *Interfaces*, 29(2):89–103, 1999.

[12] S.D. Lapierre, A. Ruiz, and P. Soriano. Designing distribution networks: formulations and solution heuristic. *Transportation Science*, to appear, 2003.

[13] C.Y. Lee. A cross decomposition algorithm for a multiproduct-multitype facility location problem. *Computers and Operations Research*, 20:527–540, 1993.

[14] T. L. Magnanti and R. T. Wong. Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research*, 29:464–484, 1981.

[15] D. McDaniel and M. Devine. A modified Benders' partitioning algorithm for mixed integer programming. *Management Science*, 24:312–379, 1977.

[16] H. Pirkul and V. Jayaraman. Production, transportation, and distribution planning in a multi-commodity tri-echelon system. *Transportation Science*, 30:291–302, 1996.

[17] J. Pooley. Integrated production and distribution facility planning at Ault foods. *Interfaces*, 24(4):113–121, 1994.

[18] E.P. Robinson, L.L. Gao, and S.D. Muggenborg. Designing an integrated distribution system at DowBrands, Inc. *Interfaces*, 23(3):107–117, 1993.

[19] D.J. Thomas and P.M. Griffin. Coordinated supply chain management. *European Journal of Operational Research*, 94:1–15, 1996.

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