A MIP Model for Commodity Warehousing and Distribution Problem

Meysam Hosseini
Department of Science, Campus of Bijar, University of Kurdistan, Bijar, Iran
Meissam_h@aut.ac.ir

Abstract
Companies are continuously looking for ways to improve their performance and stay competitive in their markets. In order to achieve the commercial goals, companies have to pay special attention to the distribution network. Warehousing and distribution of commodities to serve customers' demand are important tasks in supply chain network. The effective and successful management of the distribution network results in meeting the customers' needs directly or indirectly from plants. In this paper, a Mixed Integer Programming (MIP) formulation is proposed for Commodity Warehousing and Distribution Problem. The model determined the location of the commodities to be accumulated and decided how customer should be served while minimizing overall costs. In this network under certain condition, the customers can receive their needs directly from plants or warehousing, and each node can rule retailers and/or wholesalers. Also, a solution algorithm based on Benders decomposition was described to solve the problem. The algorithm performance was promising and computational experimentation revealed that, for randomly generated problems, the use of such integer programs helped greatly in obtaining good quality solutions.

Keywords: Mixed Integer Programming, Commodities, Warehousing, Distribution, Benders decomposition

Introduction

Trends such as globalisation, diversity of products and growing customer awareness make the markets highly competitive, thereby forcing business enterprises to adopt different strategies (Panicker et al, 2014). Ambrosino and Grazia (Ambrosino and Grazia,2005) mentioned that "all companies that aim to be competitive on the market have to pay attention to their organization related to the entire supply chain". A Supply Chain may be considered an integrated process in which a group of several organizations, such as suppliers, producers, warehouses/distributors, customers and reverse centers, work together to acquire raw materials with a view to converting them into end products which they distribute to customers (Fig. 1). As of the beginning of 1990s the concept of Supply Chain began to emerge as one of the most popular field of research and study until today (Shahroudi and Soltani, 2013). A large amount of optimization models and algorithms have been developed to make different decisions along the supply chain (Cintron, 2010).

As per the functional classification, there are four major decision areas namely procurement, manufacturing, distribution and logistics (Panicker et al, 2014). Distribution network design optimization is one highly researched area in supply chain optimization. The parties considered at this area are: the plants, the warehouse/distribution centers (DC's) and the customers (retailers and wholesalers). Distribution networks are considered as the main profitability key because they directly affect both the cost of supply chains and satisfaction of the customers. This area can be so wide and is interpreted in different ways by different scholars. It may include models that predict the number of warehouses and plants needed their locations, the production rates and inventory levels needed in these plants and warehouses, optimal routings when distributing demand, and other distribution decisions (Cintron, 2010).

Model formulations and solution algorithms which address this issue vary widely in terms of fundamental assumptions, mathematical complexity and computational performance. Some researchers have carried out their research on the base of number of warehouses required in any region. Some researchers are related to the products level and warehouse stock, while other try to optimize the sending programs. Yet some researches study the specialized applications. In most of the researches, determining the factories and warehouses required in a region is considered while designing the distribution network (Asghari Zadeh and Razani, 2013).

The problems of locating facilities and allocating customers cover the core topics of distribution system design. Model formulations and solution algorithms which address the issues vary widely in terms of fundamental assumptions, mathematical complexity and computational performance. Klose and Drexl (Klose and Drexl, 2005) emphasized that designing the distribution system are a strategic issue for almost every company. They reviewed some of the contributions to the current state-of-the-art. Tuzkaya and Onut (Tuzkaya and Onut, 2009) addressed the warehousing and transportation network design problem that involved determining the best strategy for distributing the sub-products from the suppliers to the warehouse and from the warehouse to the manufacturers. In this study, a multi-supplier, single warehouse and multi-manufacturer system was considered as an integrated warehousing and transportation network. Baker and Canessa (Baker and Canessa, 2009) presented a literature review on the overall methodology of warehouse design along with a discussion on the tools and techniques used for specific areas of analysis.

There are two key decisions when designing a distribution network: 1) Will products be delivered to the customer location or picked up from a pre-ordained site? 2) Will products flow through an intermediary? Based on the choices for the two decisions, there are various types of distribution network designs that may be used to move products from manufacturing plants to customer. Most companies have several manufacturing plants producing different products. These plants supply the costumers in the different regions according to the region's demand. However, when wholesalers or retailers have large demands they may be able to receive their products directly from these plants. For example, a policy of some companies is that if a wholesaler/retailer can fill a container from products of one plant, they can receive directly from that plant. Distributing directly from the plant is cheaper since no storage costs at the DC's are incurred. Distributor centers (DC's) buy products from the plants at a discounted price and store them in a warehouse to supply the customers. Distributors are mostly used to supply customers with low demand, for example, a family's mini-market. The main advantage of supply customers via DCs is that it can provide a faster response. The major disadvantage is the increased inventory and facility costs (Selim and Ozkarahan, 2008).

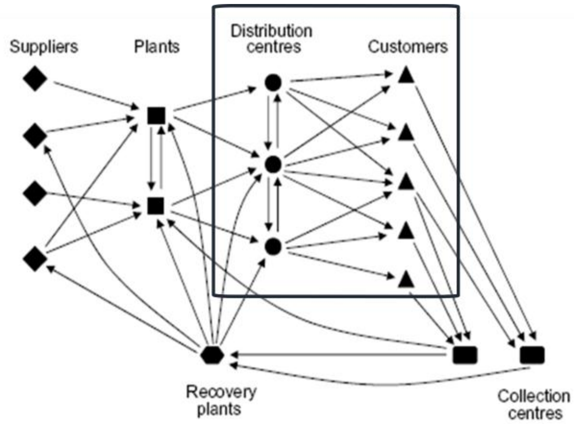

Fig. 1. A generic supply chain network

Fig. 1. depicts a generic supply chain network that includes both forward and reverse activities. In addition to different types of facilities, the possible flow of material is shown in the figure. In contrast to classical location problems, flows between facilities of the same layer are prevalent in many supply chains. These flows are usually necessary for material balancing or inventory consolidation (Melo et al, 2009). In general, planning models can be broadly categorized to three temporal classifications based on the addressed time horizons, namely strategic (long-term), tactical (mid-term) and operational (short-term). A discussion of their features and characteristics from a practical perspective was provided in (Shobrys and White, 2000).

The focus of this work is on the short-term operational planning (see Fig. 2). Operational planning is the day-by-day and month-by-month planning for what an organization is doing. It translates a high level strategic plan into a more detailed plan of who will do what and when. The corresponding model must satisfy all the operational constraints, such as mass balances, distribution constraints, customer's demand and storage requirements. Each DCs node is able to play both roles of storing and distributing the goods. So, they are restricted to their capacity as a service point and to their demand as a customer. Each node is able to support its customer as well as any demand coming from other nodes subject to the level of its storage. Its shortage can be covered by other nodes with an extra cost up to the link line limitation. In this model, it is tried to increase the relationship between costumer and producer. The model is used to determine the optimal inventory policies and production flow that satisfy demands while minimizing the total operational cost. Therefore, a customer can receive commodities directly from its known DC or indirectly from others DCs. In spite of a good progress in modeling issue, solving the new model for large size network are difficult. Therefore, a solution approach based on Benders decomposition is proposed that provides both lower and upper bounds. The technique is combining valid inequality in master problem to accelerate the Benders procedure and generate feasible solutions in sub-problems.

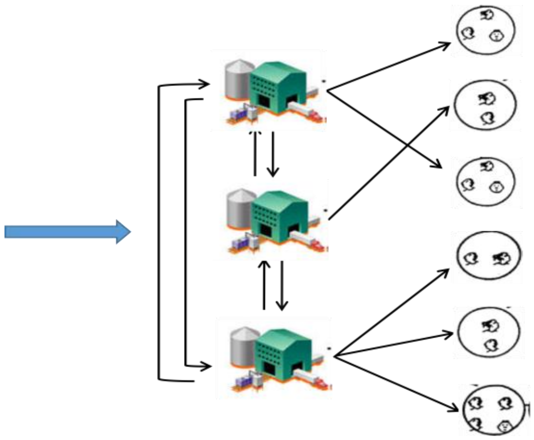

Fig. 2: The district of operational decisions

The paper is organized as follows. In the next section, the problem definition and formulation are presented in general. In the third section, the solution algorithm is detailed based on Benders decomposition. In Section 4, the experimental result is reported. Conclusions are discussed in the last section.

Model Formulation

This section presents a model includes the option of having multiple DC's in different locations and considers capacity for DC's to select the best way of configuring the existing customers so that profit is minimize cost. However, the distribution network design resulting from this model includes the location and capacity of warehouses. This model considers an optimal design as that which minimizes the distribution costs. The distribution costs include: transportation costs, inventory holding costs and storage costs. This is a major decision that companies have to make when they start or as they grow into new or larger markets. In this study, customers are retailers and wholesalers. The model formulation described here is based on above mentioned problem and a few assumptions are taken into account. The main assumptions of the optimization model are as follows:

A1: Each node is able to store commodities as a storage point and serves its customer as a DC
A2: A node is connected to other nodes with a direct link and can support others
A3: A node can demand for a commodity if its customer requests
A4: The capacity of nodes and links are known and limited
A5: The system can serve for more than one commodity

Here, the problem statement and model formulation are progressively explained. A mesh network is presented by the undirected graph $G=<V,E>$, where V is a set of n nodes and E is the set of arcs corresponding to $n(n-1)/2$ links, respectively. The following notation is used for the sets, parameters and variables:

Name	Description
Set and Indices	
V	The set of nodes $\{1,2,...,n\}, i, j \in V$
P	The set of commodities $\{1,2,...,m\}, k \in P$
E	The set of arcs
Parameters	

S_j	The storage capacity at node $j \in V$
e_{ij}	1 if there is a link connecting nodes i and j
S_{ij}	The passing capacity at link e_{ij}
b_j^k	The demand for commodity $k \in P$ in node $j \in V$
c_j^k	The cost of storing commodity $k \in P$ in node $j \in V$
c_{ij}^k	The cost of transferring commodity $k \in P$ from node $i \in V$ to node $j \in V$ via link e_{ij}
Variables	
x_{ij}^k	1 if commodity $k \in P$ is moved from node $i \in V$ to node $j \in V$
y_j^k	1 if commodity $k \in P$ is stored in node $j \in V$

Using this model, decision was made on how a node must store goods and serve its customer based on available commodities, at stock or obtainable from other nodes subject to operational restrictions and capacity limitations. The overall aim was to minimize total storage and transportation cost while satisfying total demand. The following integer linear programming formulation was considered for Commodity Warehousing and Distribution Problem.

$$(F) \quad \min \quad \sum_{k \in P} \sum_{j \in V} \bar{c}_j^k y_j^k + \sum_{k \in P} \sum_{j \in V} \sum_{i \in V, i \neq j} \bar{c}_{ij}^k x_{ij}^k \tag{1}$$

s.t.

$$\sum_{k \in P} b_j^k y_j^k + \sum_{k \in P} \sum_{\substack{i \in V \\ i \neq j}} b_i^k x_{ji}^k \leq S_j \qquad \forall j \in V \tag{2a}$$

$$\sum_{k \in P} b_j^k x_{ij}^k \leq S_{ij} \qquad \forall i \neq j \in V \tag{3a}$$

$$\sum_{i \in V, i \neq j} x_{ij}^k + y_j^k = 1 \qquad \forall j \in V, k \in P \text{ if } \quad b_j^k > 0 \tag{4}$$

$$x_{ij}^k \leq y_i^k \qquad \forall i \neq j \in V, k \in P \tag{5}$$

$$x_{ij}^k, y_j^k \in \{0,1\} \qquad \forall i \neq j \in V, k \in P \tag{6}$$

where $\bar{c}_j^k = c_j^k b_j^k$ and $\bar{c}_{ij}^k = c_{ij}^k b_j^k$. Constraints $(2a),(3a)$ are forcing storage capacity in nodes and arcs, respectively. Constraint (4) states that each node either stores a commodity or receives it from another node if the node has demand. Constraint (5) implies that a commodity may be only obtained from a node if it is stored there. Constraint (6) imposes integrality restrictions on the variables.

If $\forall j \in V, k \in P; b_j^k > 0$, then problem F has $(n^2 m + nm)/2$ binary variables and $(n^2 m + n^2 + m)/2$ constraints, where n is the number of nodes, $n = |V|$, and m the number of commodities, $m = |P|$.

There are many applications of this problem in different industries. For example, see (Gollowitzer and Ljubic, 2011) for connected facility location Rent-or-buy problem, (Vlachos and Iakovou, 2005), (Farahani and Grunow, 2010) for food distribution and for video placement and routing problem (VPRP)

(Ouveysi et al, 2002). The VPRP, as a special case of the model, belongs to NP-hard problems. Therefore, problem F, in general, is an NP-hard problem, so an optimum solution for F is not practical using standard software, especially by increasing the problem size. So, a solution algorithm for this problem is introduced here which is based on Benders decomposition that greatly helps in obtaining good quality solutions.

Solution Methods Based on Benders Decomposition

In this section, a solution algorithm for CWDP is presented based on Benders decomposition. Then, a modified version of algorithm is introduced to overcome the difficulties due to the problem size. In 1962, Benders (Benders, 1962) proposed a partitioning algorithm for solving specially-structured large-scale linear and mixed integer programs. As applied to mixed-integer programming, Benders' original work made two primary contributions: (1) development of a "pure integer" problem that is equivalent to the original mixed-integer problem, and (2) a relaxation algorithm for solving problem that works iteratively on an LP problem and a "pure integer" problem.

A good review of the solution methods based on Benders decomposition was given by (Costa, 2005) in which it was shown that the number of researchers using this technique has been increasing. The main idea was to decompose a large-scale MIP problem to a master problem (small IP) and some sub-problems (large LP) in order to derive an equivalent master problem by generating Benders feasibility and optimality cuts as prompted by the sub-problems. Here, the method is rewritten as illustrated in (Canto, 2008).

Consider the MIP problem: $\min\{\, c^T x + f^T y \mid Ax + By \ge b, Dy \ge h,\ x \ge 0\,\}$. Vectors x and y are the continuous and integer variables, respectively. If y is fixed at a feasible integer configuration $\bar{y}$, the resulting model is: $\min_x \{\, c^T x \mid Ax \ge b - B\bar{y},\ x \ge 0\}$ and the complete minimization problem can, therefore, be written as: $\min_{y \in Y} \left[f^T y + \min_{x \ge 0} \left\{ c^T x \mid Ax \ge b - By \right\} \right]$ where $Y = \{\, y \mid Dy \ge h\,\}$. The dual of the inner LP problem is $\max_\pi \{\ (b - B\bar{y})^T \pi \mid A^T \pi \le c,\ \pi \ge 0\}$, where π is variable of dual problem. In any iterations of the Benders' decomposition algorithm, two different problems are solved. A restricted master problem which in the form of:

$$\min_{y \in Y} \{ z \mid z \ge f^T y + (b - By)^T \bar{\pi}_l \ \ l = 1, \ldots, L,\ (b - By)^T \bar{\pi}_l \le 0, l = 1, \ldots, L \}$$

and a sub-problem in the form of: $\max_\pi \{ f^T \bar{y} + (b - B\bar{y})^T \pi \mid A^T \pi \le c,\ \pi \ge 0\}$ where $\bar{\pi}_l$ is the solution to the sub-problem on the l^{th} iteration.

Now, problem F' is considered with slack variables u and w.

$$(F') \quad \min \sum_{k \in P} \sum_{j \in V} \bar{c}_j^k y_j^k + \sum_{k \in P} \sum_{j \in V} \sum_{i \in V, i \ne j} \bar{c}_{ij}^k x_{ij}^k + \sum_{j \in V} \sum_{i \in V, i \ne j} M \left(u_j + w_{ij} \right)$$

s.t.

$$\sum_{k \in P} b_j^k y_j^k + \sum_{k \in P} \sum_{\substack{i \in V \\ i \ne j}} b_i^k x_{ji}^k \le S_j + u_j \qquad \forall j \in V \tag{2b}$$

$$\sum_{k \in P} b_j^k x_{ij}^k \le S_{ij} + w_{ij} \qquad \forall i \ne j \in V \tag{3b}$$

$$\sum_{i \in V, i \ne j} x_{ij}^k + y_j^k = 1 \qquad \forall j \in V, k \in P \mid d_j^k = 1$$

$$x_{ij}^k \le y_i^k \qquad \forall i \ne j \in V, k \in P$$
$$x_{ij}^k, y_j^k \in \{0,1\} \quad , \quad u_j, w_{ij} \ge 0 \qquad \forall i \ne j \in V, k \in P$$

Where M is a big positive number. The first master problem, BF, can be obtained by removing capacity constraints $(2b)$ and $(3b)$.

$$(BF) \quad \min \sum_{k \in P} \sum_{j \in V} \bar{c}_j^k y_j^k + \sum_{k \in P} \sum_{j \in V} \sum_{i \in V, i \ne j} \bar{c}_{ij}^k x_{ij}^k$$

s. t.

$$\sum_{i \in V, i \ne j} x_{ij}^k + y_j^k = 1 \qquad \forall j \in V, k \in P \mid d_j^k = 1$$
$$x_{ij}^k \le y_i^k \qquad \forall i \ne j \in V, k \in P$$
$$x_{ij}^k, y_j^k \in \{0,1\} \qquad \forall i \ne j \in V, k \in P$$

Let x_{ij}^{k*} and y_j^{k*} be the optimal solution for BF. If this solution is feasible for problem F, then the optimal solution of the original problem is obtained, that is equivalent to the zero optimal objective value of the sub-problems $P1$ and $P2$ (or, $DP1$ and $DP2$); otherwise, the following sub-problems, P_1, P_2 or their duals DP_1, DP_2 are solved:

$P1 \quad \min \sum_{j \in V} u_j$ $s.t. \quad u_j \ge \sum_{k \in P} b_j^k y_j^{k*} + \sum_{k \in P} \sum_{i \in V, i \ne j} b_i^k x_{ji}^{k*} - S_j \qquad \forall j \in V$ $u_j \ge 0 \qquad \forall j \in V$	$P2 \quad \min \sum_{j \in V} \sum_{i \in V, i \ne j} w_{ij}$ $s.t. \quad w_{ij} \ge \sum_{k \in P} b_j^k x_{ij}^{k*} - S_{ij} \qquad \forall i \ne j \in V$ $w_{ij} \ge 0 \qquad \forall i \ne j \in V$
$DP1 \quad \max \sum_{j \in V} \pi_j \left(\sum_{k \in P} b_j^k y_j^{k*} + \sum_{k \in P} \sum_{i \in V, i \ne j} b_i^k x_{ji}^{k*} - S_j \right)$ $s.t. \quad \pi_j \le 1 \qquad \forall j \in V$ $\pi_j \ge 0 \qquad \forall j \in V$	$DP2 \quad \max \sum_{j \in V} \sum_{i \in V, i \ne j} \pi'_{ij} \left(\sum_{k \in P} b_j^k x_{ij}^{k*} - S_{ij} \right)$ $s.t. \quad \pi'_{ij} \le 1 \qquad \forall i \ne j \in V$ $\pi'_{ij} \ge 0 \qquad \forall i \ne j \in V$

A non-zero optimal objective value to these sub-problems is corresponding to the infeasibility in problem F. So, the feasibility cuts (7, 8) have to be added to BF.

$$\sum_{j \in V} \pi_j^* \left(\sum_{k \in P} b_j^k y_j^k + \sum_{k \in P} \sum_{i \in V, i \ne j} b_i^k x_{ji}^k - S_j \right) \le 0 \qquad (7)$$

and

$$\sum_{j \in V} \sum_{i \in V, i \ne j} \pi_{ij}'^* \left(\sum_{k \in P} b_j^k x_{ij}^k - S_{ij} \right) \le 0 \qquad (8)$$

The special structure of dual of sub-problems, $DP1$ and $DP2$ shows that $\pi_j^* = \max\{0, sign(\overline{S_j})\}$, $\pi_{ij}'^* = \max\{0, sign(\overline{S_{ij}})\}$ are the optimum solution where $\overline{S_j} = \sum_{k\in P} b_j^k y_j^{k*} + \sum_{k\in P}\sum_{\substack{i\in V\\ i\neq j}} b_i^k x_{ji}^{k*} - S_j$ and $\overline{S_{ij}} = \sum_{k\in P} b_j^k x_{ij}^{k*} - S_{ij}$.

In the case of having multi optimal solutions for sub-problems in (Magnanti and Wong, 1981) and (Papadakos, 2008) proposed to generate Pareto optimal cuts. Also the cut density is another concern that a remedy such as the covering cut bundle generation method proposed in (Saharidis et al, 2010). In our case the optimality cut is not applicable and only feasibility is concern because any feasible solution will be optimum. Moreover the number of required iterations and the way we calculate optimum solutions suggest that the straightforward approach based on original Benders Decomposition is efficient. Our experiments with different randomly generated problems are also shown that the difference is not significant. Also adding valid inequalities as is recommended، by researchers (Saharidis et al, 2011), (Cordeau et al, 2000 and 2006) and (Andreas and Smith, 2009) is not helpful in our case.

The new problem, consisting of original constraints and feasibility cuts, is set out in BF_L where $\left(\pi_j^*\right)^l$ and $\left(\pi_{ij}'^*\right)^l$ are optimal solutions of dual problems DP_1, DP_2 in the l^{st} iteration, respectively.

$$BF_L \quad \min \sum_{k\in P}\sum_{j\in V} \bar{c}_j^k y_j^k + \sum_{k\in P}\sum_{j\in V}\sum_{i\in V, i\neq j} \bar{c}_{ij}^k x_{ij}^k$$

s. t.

$$\sum_{i\in V, i\neq j} x_{ij}^k + y_j^k = 1 \qquad \forall j\in V, k\in P \mid d_j^k = 1$$

$$x_{ij}^k \le y_i^k \qquad \forall i\neq j\in V, k\in P$$

$$\sum_{j\in V}\left(\pi_j^*\right)^l \left(\sum_{k\in P} b_j^k y_j^k + \sum_{k\in P}\sum_{\substack{i\in V\\ i\neq j}} b_i^k x_{ji}^k - S_j\right) \le 0 \quad \forall \ l = 1,2,\ldots,L$$

$$\sum_{j\in V}\sum_{i\in V, i\neq j}\left(\pi_{ij}'^*\right)^l \left(\sum_{k\in P} b_j^k x_{ij}^k - S_{ij}\right) \le 0 \qquad \forall \ l = 1,2,\ldots,L$$

$$x_{ij}^k, y_j^k \in \{0,1\}, \qquad \forall i\neq j\in V, k\in P$$

Starting from a given $\varepsilon > 0$ and max iteration, now the optimization algorithm is:

Starting from a given $\varepsilon > 0$ and max iteration, now the optimization algorithm is:

1. Let $Z_{BF^0} = 0$ and $l = 1$;
2. Perform the following steps until $gap = \dfrac{Z_{BF^l} - Z_{BF^{l-1}}}{Z_{BF^l}} \le \varepsilon$ or $\left(\pi_j^*\right)^l$ and $\left(\pi_{ij}'^*\right)^l$ are zero or $l > Max\ iteration$; where Z_{BF^l} is the best solution for the BF problem in l iterations.
    a. Solve BF problem and obtain optimal solution x_{ij}^{k*} , y_j^{k*} .

b. Calculate $\left(\pi_j^{\ *}\right)^l$, $\left(\pi_{ij}^{\prime\ *}\right)^l$, If $\left(\pi_j^{\ *}\right)^l$ and $\left(\pi_{ij}^{\prime\ *}\right)^l$ are zero, so the current solution is feasible and optimal: otherwise,

c. Add new constraints (7) and/or (8) to BF problem.

d. Increment l by 1 and then go to (a).

3. Output solution x_{ij}^{k*} , y_j^{k*} as the best solution;

It is worth mentioning that any solution for master problem BF_L which is feasible to the original problem F would be an optimal solution.

A Modified Algorithm

The experimental result in Table 1 shows that, using Benders decomposition, a lower bound with higher accuracy can be obtained. Unfortunately, the algorithm efficiency decreases with increasing the problem size. The following modifications were made to overcome this shortcoming. The modifications included solving LP-relaxation of master problem BF and then following some corrections, the method has been considered by McDaniel and Devine (McDaniel and Devine, 1977) as well as Goetschalckx and Dogan (Goetschalckx and Dogan, 1999).
The lower bound obtained by this way was surely less than the lower bound obtained by the original algorithm, but solving linear programs instead of integer programs at any iteration was expected to help in speeding up the algorithm. The only difficulty of this modification was that the optimal solution of the LP-relaxation was in general fractional, but this was not always the case. A heuristic method was employed for obtaining feasible solutions. However, this situation could be fixed through rounding up (to 1) every fractional variable with a value greater or equal to 0.5, and rounding down (to 0) the rest of them. Then, the obtained solution could be modified to a feasible one.

Let $\hat{y}_j^{\,k}$ and $\hat{x}_{ij}^{\,k}$ be rounded values corresponding to the optimum solution of relaxed problem. The rounding process has no affect on the feasibility of constraints (5). Now, Constraints in (4) are verified; for $j \in V, k \in P$ *that* $b_j^k > 0$ if this constraint does not satisfy after being rounded, two cases may occur; either $\sum_{i \in V, i \neq j} \hat{x}_{ij}^{\,k} + \hat{y}_j^{\,k} > 1$ or $\sum_{i \in V, i \neq j} \hat{x}_{ij}^{\,k} + \hat{y}_j^{\,k} = 0$. In any case, set $\hat{y}_j^{\,k} = 1$ and $\hat{x}_{ij}^{\,k} = 0$, $\forall k \in P, j \in V$. The new solution satisfy both Constraints $(4),(5)$ and the algorithm can be continued from step (2-b).

An Application and Numerical Results

As a practical case, the above approach was applied on VPRP where a node is a server which is able to store a copy of program (says k) in which λ_k is the capacity requirement for storing program $k \in P$ at any node and the bandwidth requirement for transmission of program $k \in P$ in the network using μ_k from its link capacity. Also, there is a demand for each program at each node, see (Ouveysi et al, 2002). So, a server plays two roles as a warehouse and DC. As a result, Constraint (2a) and (3a) are

$$\sum_{k \in P} \lambda_k y_j^{\,k} \le S_j \qquad \forall j \in V$$

$$\sum_{k \in P} \mu_k x_{ij}^k \le S_{ij} \qquad \forall i \neq j \in V$$

In this case, computer programs were considered as different commodities; therefore, one copy of a program was enough for supporting many requests. Consequently, the second term in Constraint (2a) was not necessary and removed. In this section, the present computational experiment is described with the proposed algorithm using randomly generated test problems for VPRP.

To compare the performance of different approaches, a batch of 16 random problems were generated with the number of nodes (n) ranging from 50 to 80 and the number of commodities (m) ranging from 20 to 50, as described in (Bektas et al, 2007). Parameters μ_k, c_{ij}^k, c_j^k were randomly generated from a continuous uniform distribution between 50 and100. λ_k was modeled as $\lambda_k = \mu_k T_k$ where T_k is the total transmission time for program k. In the experiments, $T_k = 10$ fixed for all $k \in P$. The S_{ij} values were chosen from the continuous uniform distribution between $\max_{k \in P}\{\mu_k\}$ and $\sum_{k \in P} \mu_k$. The capacity of each node (S_j) was set to be 40% of the size of all the programs. The proposed algorithm was implemented using AIMMS software (AIMMS, 3.9) and all the test problems were solved on a PC Intel processor 2.4 GHz and 3.00 Gb of RAM using CPLEX 12.0 (CPLEX, 12.0) as the optimization package.

In order to compare the performance of different approaches, a time limit of 500 sec was considered. The best objective values of above-mentioned approaches within the given time limit were denoted by v_C, v_L and v_B, respectively.

The computational results are shown in Table 1. Any row in Table 1 consists of:

n : Number of nodes;

m : Number of commodities;

n_B : Number of iterations required by the Benders algorithm;

n_L : Number of iterations required by the Lagrangean algorithm;

T_B : Solution time using Benders algorithm (sec);

T_L : Solution time using Lagrangean algorithm (sec);

T_C : Solution time required to solve instances using CPLEX (sec);

$g_B = (\frac{v_C - v_B}{v_C}) \times 100$: Relative gap (error%);

$g_L = (\frac{v_L - v_C}{v_C}) \times 100$: Relative gap (error%)

The results in Table 1 indicate that the proposed algorithm is able to produce good quality solutions. The algorithm delivered a high-quality near-optimal solution in all cases (about 1% error). The quality of the other solution was not as good as that of the proposed algorithm, even with an increase in time and no. of iterations. Only, in two cases, the solution of Lagrangean approach was better than the proposed algorithm. The solution times of the proposed algorithm for some instances (special for large instances) are better than the solution time of Lagrangean approach.

Table 1: Comparison of different approaches in terms of solution quality

#	n	m	n_B	n_L	T_B	T_L	T_C	v_B	v_C	g_B	g_L
1	50	20	6	13	66	62.23	57.31	53885.00	53891	0.0111	0.82
2	50	30	6	16	161	184.56	126.07	80954.11	80956	0.0023	0.98

3	50	40	6	10	245	212.32	188.53	107627.50	107634	0.0060	0.53
4	50	50	6	9	327	328.65	302.20	134653.00	134660	0.0051	-0.10
5	60	20	6	12	134	84.25	121.63	64036.42	64041	0.0071	0.91
6	60	30	6	10	326	323.65	264.29	96175.41	96179	0.0037	0.72
7	60	40	6	7	475	456.56	322.63	128597.70	128602	0.0033	-0.56
8	60	50	6	7	500	485.24	476.92	160391.60	160400	0.0052	-2.61
9	70	20	2	7	420	436.25	126.87	74518.00	74518	0.00	0.12
10	70	30	6	7	430	473.98	371.25	111701.70	111707	0.0047	-2.74
11	70	40	6	7	500	500.00	442.32	148852.40	148854	0.0011	-3.63
12	70	50	6	6	500	500.00	490.41	186155.80	186159	0.0017	-3.89
13	80	20	6	6	390	387.45	154.27	84846.56	84848	0.0016	-0.30
14	80	30	6	7	485	500.00	365.85	127058.90	127062	0.0024	-5.08
15	80	40	6	8	500	500.00	479.29	169489.10	169491	0.0011	-4.80
16	80	50	6	7	500	500.00	500	211905.00	211914	0.0042	-4.07

In facing with large scale problems, the size of master problem increases dramatically and obtaining IP solution gets more difficult. The modified algorithm was proposed to overcome this shortcoming. The aim of this approach is to improve the solution time. The performances of the modified algorithm and Benders algorithm are reported in Table 2. As can be seen, the solution time for the modified algorithm is better than Benders algorithm.

Therefore, 20 random problems were generated with the number of nodes (n) ranging from 50 to 90 and the number of commodities (m) ranging from 10 to 40. The best objective value of the two approaches was denoted by v_B and v_M , respectively. Any row in Table 2 consists of:

n : Number of nodes;

m : Number of commodities;

n_B : Number of iterations required by Benders algorithm;

n_M : Number of iterations required by modified algorithm;

T_B : Solution time using Benders algorithm (sec);

T_M : Solution time using modified algorithm (sec.);

T_C : Solution time required to solve instances using CPLEX (sec);

$imp = \frac{T_B - T_M}{T_B} \times 100$: Relative time improvement;

gCplex: final gap of the best solution found by CPLEX using time limit (%);

$g_M = \frac{v_B - v_M}{v_B} \times 100$: Relative gap (error%);

$g_B = (\frac{v_C - v_B}{v_C}) \times 100$: Relative gap (error%);

$g_{CM} = \frac{v_C - v_M}{v_C} \times 100$: Relative gap (error%).

Table 2: Comparison of Benders and modified algorithms in terms of solution time and quality

#	n	m	n_B	n_M	T_B	T_M	T_C	imp	v_B	v_M	v_C	g_{Cplex}	g_M	g_B	g_{CM}
1	50	10	6	6	16	6	11.3	62.5	26897.67	26888	26901	0	0.0359	0.0123	0.0483
2	50	20	6	6	60	12	57.31	80	53900.34	53896	53907	0	0.0080	0.0123	0.0204

3	50	30	6	6	166	17	126.07	89.759	80687	80669	80694	0	0.0223	0.0086	0.0309
4	50	40	6	6	224	28	188.53	87.5	107811.7	107778	107820	0	0.0312	0.0077	0.0389
5	60	10	6	6	46	7	12.51	84.782	32127	32110	32133	0	0.0529	0.0186	0.0715
6	60	20	6	6	138	17	121.63	87.681	64163	64149	64170	0	0.0218	0.0109	0.0327
7	60	30	6	6	254	25	264.29	90.157	96313	96294	96314	0	0.0197	0.0010	0.0207
8	60	40	6	6	506	30	322.63	94.071	128128	128109	128133	0	0.0148	0.0039	0.0187
9	70	10	6	6	107	12	31.74	88.785	37247	37238	37252	0	0.0241	0.0134	0.0375
10	70	20	6	6	331	33	126.87	90.030	74574	74551	74580	0	0.0308	0.0080	0.0388
11	70	30	6	6	389	54	318.25	86.118	111610	111591	111611	0	0.0170	0.0008	0.0179
12	70	40	6	6	605	64	442.32	89.421	148938	148914	148944	0	0.0161	0.0040	0.0201
13	80	10	6	6	128	22	39.30	82.812	42348	42334	42349	0	0.0330	0.0023	0.0354
14	80	20	6	6	322	38	154.27	88.198	84732	84721	84732	0	0.0129	0.00	0.0129
15	80	30	6	6	494	62	365.85	87.449	127058.5	127034	127062	0	0.0192	0.0027	0.0220
16	80	40	6	6	518	92	419.29	89.978	169407.8	169377	169410	0	0.0181	0.0013	0.0194
17	90	10	6	6	312	26	48.29	91.666	47627	47610	47630	0	0.0356	0.0062	0.0419
18	90	20	3	6	280	54	294.39	80.714	94967	94955	94971	0	0.0126	0.0042	0.0168
19	90	30	3	6	518	143	500	84.422	142478.7	142422	142489	0	0.0398	0.0072	0.0470
20	90	40	3	6	500	257	500	82.866	190407	190354	190420	0.02%	0.0278	0.0068	0.0346

The results reported in Table 2 show that applying the modified algorithm could lead to a reduction in solution time up to 80% while keeping solution quality at the same level. Also, the large-scale problems were solved efficiently.

Conclusion

In this paper, the warehousing and distribution of plant's commodities was proposed in which demands of customers were served while minimizing the total cost of storage and transportation commodities such that facility and link capacity were satisfied. The MIP model as a Commodity Warehousing and Distribution Problem is NP-hard. So, a modified Benders decomposition algorithm was presented to get an optimal solution. It is clear that obtaining the optimal solution for the considered model gets more complicated when the problem size increases. The computational result showed that the proposed algorithm is able to obtain near-optimal solutions in a reasonable time even for large-scale problems. The decomposition algorithm takes the problem structure into account and looks for a feasible solution in any iteration. Therefore, only feasibility cuts are added to the master problem. This feature prevents the master problem from quickly getting large and leads to a reduction in the solution time.

References

AIMMS. (3.9). Paragon Decision Technology B.V., Netherlands.

Akkerman, R., Farahani, P., & Grunow, M. (2010). Quality, safety and sustainability in food distribution: a review of quantitative operations management approaches and challenges. OR Spectrum , 32, 863–904.

Ambrosino, D., & Grazia, M. (2005). Distribution network design: New problems and related model. European Journal of Operational Research , 165 (3), 610 – 624.

Andreas, A. K., & Smith, J. C. (2009). Decomposition algorithms for the design of a nonsimultaneous capacitated evacuation tree network. Networks , 53 (2), 91-103.

Asghari Zadeh, E., & Razani, E. (2013). Designing a Product Distribution Network Using Mixed Integer Planning. British Journal of Science , 7 (2), 64-79.

Baker, P., & Canessa, M. (2009). Warehouse design: A structured approach. European Journal of Operational Research , 193, 425–436.

Bektas, T., Oguz, O., & Ouveysi, I. (2007). A Lagrangean relaxation and decomposition for the video placement and routing problem. European Journal of Operational Research , 182, 455–465.

Benders, B. (1962). Partitioning procedures for solving mixed-variables programming problems. Numerisch Mathematik 4 , 238–252.

Canto, S. P. (2008). Application of Benders' decomposition to power plant preventive maintenance scheduling. European Journal of Operational Research , 184 (2), 759-777.

Cintron, A. L. (2010). OPTIMIZING AN INTEGRATED SUPPLY CHAIN. The Pennsylvania State University: The Graduate School.

Cordeau, J. F., Pasin, F., & Solomon, M. M. (2006). An integrated model for logistics network design. Annals of Operations Research , 144 (1), 59–82.

Cordeau, J. F., Soumis, F., & Desrosiers, J. (2000). A Benders decomposition approach for the locomotive and car assignment problem. Transportation Science , 34 (2), 133-149.

Costa, A. M. (2005). A survey on benders decomposition applied to fixed-charge network design problems. Computers and Operations Research , 32, 1429-1450.

CPLEX, I. (12.0). User's Manual ILOG Inc., CPLEX Division, Mountain View, USA.

Georgiadis, P., Vlachos, D., & Iakovou, E. (2005). A system dynamics modeling framework for the strategic supply chain management of food chains. Journal of Food Engineering , 70, 351–364.

Goetschalckx, M., & Dogan, K. (1999). A primal decomposition method for the integrated design of multi-period production-distribution systems. IIE Transactions , 31 (11), 1027-1036.

Gollowitzer, S., & Ljubic, I. (2011). MIP models for connected facility location: A theoretical and computational study. Computers and Operations Research , 38, 435–449.

Klose, A., & Drexl, A. (2005). Facility location models for distribution system design. European Journal of Operational Research , 162, 4–29.

Magnanti, T. L., & Wong, R. T. (1981). Accelerating Benders decomposition: algorithmic enhancement and model selection criteria. Operations Research , 29 (3), 464-484.

McDaniel, D., & Devine, M. (1977). A modified Benders partitioning algorithm for mixed integer programming. Management Science , 24, 312-319.

Melo, M. T., Nickel, S., & Saldanha-da-Gama, S. (2009). Facility location and supply chain management – A review. European Journal of Operational Research , 196, 401–412.

Ouveysi, I., Sesana, L., & Wirth, A. (2002). The video placement and routing problem. Operations Research/Management Science at Work , 43 (Kluwer Academic Publishers), 53-71.

Panicker, V. V., Vanga, R., & Sridharan, R. (2014). Ant colony optimisation algorithm for distribution-allocation problem in a two-stage supply chain with a fixed transportation charge. International Journal of Production Research , 51 (3), 698–717.

Papadakos, N. (2008). Practical enhancements to the Magnanti-Wong method. Operations Research Letters , 36 (4), 444-449.

Saharidis, G. K., Boile, M., & Theofanis, S. (2011). Initialization of the Benders master problem using valid inequalities applied to fixed-charge network problems. Expert Systems With Applications , 38 (6), 6627-6636.

Saharidis, G. K., Minoux, M., & Ierapetritou, M. G. (2010). Accelerating Benders method using covering cut bundle generation. International Transactions in Operational Research , 17 (2), 221-237.

Selim, H., & Ozkarahan, I. (2008). A supply chain distribution network design model: An interactive fuzzy goal programming-based solution approach. Int J Adv Manuf Technol , 36, 401–418.

Shahroudi, K., & Soltani, S. (2013). Application of goal programming for target setting in Irans auto industry. Int. J. Industrial Mathematics , 5 (3), 205-216.

Shobrys, D. E., & White, D. C. (2000). Planning, scheduling and control systems: why can they not work together. Comput. Chem. Eng. , 24, 163–173.

Tuzkaya, U. R., & Onut, S. (2009). A holonic approach based integration methodology for transportation and warehousing functions of the supply network. Computers and Industrial Engineering , 56, 708–723.