Transportation Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

A Transportation Problem with Minimum Quantity Commitment

Andrew Lim, Fan Wang, Zhou Xu,

TRANSPORTATION SCIENCE
Vol. 40, No. 1, February 2006, pp. 117–129
ISSN 0041-1655 | EISSN 1526-5447 | 06 | 4001 | 0117

DOI 10.1287/trsc.1050.0123

A Transportation Problem with Minimum Quantity Commitment

Andrew Lim, Fan Wang, Zhou Xu*
Department of Industrial Engineering and Engineering Management, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

We study a transportation problem with the minimum quantity commitment (MQC), which is faced by a famous international company. The company has a large number of cargos for carriers to ship to the United States. However, the U.S. Marine Federal Commission stipulates that when shipping cargos to the United States, shippers must engage their carriers with an MQC. With such a constraint of MQC, the transportation problem becomes intractable. To solve it practically, we provide a mixed-integer programming model defined by a number of strong facets. Based on this model, a branch-and-cut search scheme is applied to solve small-size instances and a linear programming rounding heuristic for large ones. We also devise a greedy approximation method, whose solution quality depends on the scale of the minimum quantity if the transportation cost forms a distance metric. Extensive experiments have been conducted to measure the performance of the formulations and the algorithms and have shown that the linear rounding heuristic behaves best.

Key words: logistics; minimum quantity commitment; selection and assignment; branch and cut; heuristics
History: Received: October 2003; revision received: March 2005; accepted: May 2005.

1. Introduction

Assume a set of carriers, $I = \{1, 2, \ldots, m\}$, and a set of customers, $J = \{1, 2, \ldots, n\}$. Each customer $j \in J$ has d_j cargos and will be charged a transportation cost of $c_{i,j}$ for a carrier $i \in I$ to deliver a unit cargo. The transportation costs, $c_{i,j}$ for $i \in I$ and $j \in J$, are nonnegative and symmetric. The *transportation problem* is to decide the allocation of customers' cargos to carriers so that the total transportation cost is minimized.

Let $z_{i,j}$ denote the decision variable representing the number of cargos of customer j allocated to the carrier i for $i \in I$ and $j \in J$. The objective of the transportation problem can be formulated as

$$\sum_{i\in I}\sum_{j\in J} c_{i,j} z_{i,j}, \tag{1}$$

which minimizes the total transportation cost.

Because all the cargos have to be allocated to carriers, we have the following demand constraint:

$$\sum_{i\in I} z_{i,j} = d_j, \quad \text{for } i \in I \text{ and } j \in J. \tag{2}$$

Because we can always allocate each cargo to the carrier whose transportation cost is the lowest, the original transportation problem can be efficiently solved in $O(mn)$ time.

Various practical restrictions have been studied in the literature, such as carriers' maximum capacity for the distribution problem (Ahuja, Magnanti, and Orlin 1993), carriers' fixed selection costs for the facility location problem (Jain, Mahdian, and Saberi 2002), and carriers' total number for the p-median problem (Bozkaya, Zhang, and Erkut 2002). These restrictions make the original transportation problem harder, and the latter two lead the problem to be NP-hard.

In this paper, we study another constraint from real practice that also brings intractable difficulties for the transportation problem. It is motivated by the U.S. Federal Maritime Commission, which stipulates that the total quantity of cargos delivered by each carrier to U.S. cities must either be none or at least as large as a fixed minimum quantity. For this reason, global shippers have to allocate a minimum quantity of cargos, denoted by b, to carriers when requesting carriers to ship cargos to their customers in the United States (for security reasons, we eliminate the name of the company we consulted). This stipulation is often denoted by *minimum quantity commitment* (MQC) and can be formulated as

$$\sum_{j\in J} z_{i,j} = 0, \quad \text{or} \quad \sum_{j\in J} z_{i,j} \ge b, \quad \text{for } i \in I. \tag{3}$$

Accordingly, the transportation problem with MQC aims at minimizing the total cost (1) to satisfy demands (2) and MQC (3).

To illustrate the problem further, let us consider a manufacturer who will ship cargos to two customers (Customers A and B) in the United States by two

*Corresponding author: xuzhou@ust.hk

candidate shipping carriers (e.g., DHL and UPS). The cargos demands for Customer A and Customer B are 100 units each. The unit transportation costs for Customer A are 1 by DHL and 1.5 by UPS, while the unit transportation costs for Customer B are 2 by DHL and 1 by UPS. Clearly, without considering the MQC, the optimal solution is to let DHL ship all 100 units of cargos for Customer A and UPS ship all 100 units of cargos for Customer B. However, if MQC must be considered, say $b = 150$, the above solution violates the MQC constraint because neither DHL nor UPS ships 150 units or more. Thus, for the case with MQC, the optimal solution is to let UPS ship all 200 units of cargos for both Customers A and B.

The transportation problem with MQC can be classified into two classes: the *metric version* and the *nonmetric version*. The metric version makes the restriction that the transportation costs satisfy the triangular inequality. In other words, in the metric version, $c_{i,j}$ depends on the distance between the location of carrier i and the location of customer j, which forms a metric. The metric version is practical in some real applications. However, in other situations, the transportation costs are not necessary to satisfy the triangular inequality. Without assuming transportation costs to satisfy the triangular inequalities, we have the nonmetric version of the problem, which is more general than the metric version.

This paper focuses on the nonmetric version of the transportation problem with MQC; in terms of the computational complexities, a mixed-integer programming (MIP) model with improvements, and two heuristic algorithms. One heuristic is based on the linear-rounding method, which performs accurately in our experiments. The other is a greedy heuristic, whose solution quality depends on the scale of the minimum quantity when the transportation cost forms a distance metric.

The rest of the paper is organized as follows. We summarize more applications of MQC and the literature on related topics in §2. Because the transportation problem with the MQC constraint is proved to be intractably *NP*-hard, we propose and strengthen its MIP model in §3. Based on this model, we apply a branch-and-cut algorithm in §4 to obtain exact optimal solutions and propose two efficient heuristics to solve it practically in §5. To measure the performance of different models and algorithms, extensive experiments have been conducted. Section 6 reports the experimental results and some analysis of these. Lastly, we conclude the paper in §7.

2. Literature Review

In addition to its application in transportation, the MQC is also applied in other businesses and industries. Bassok and Anupindi (1997) and Chen and Krass (1999) introduced supply contracts with minimum-total-order quantity commitments. In a simple buyer-supplier arrangement, the buyer can place an order for any amount. While this arrangement affords the buyer a great deal of flexibility, it also results in a great deal of uncertainty for the supplier. In many industries, to balance the flexibility for the buyer and the reduction of uncertainty for the supplier, a buyer and a supplier always make, in advance, a commitment-purchase contract according to which the buyer must purchase a certain number of units or more from the suppliers. In addition, Bonser and Wu (2001) studied a real case in fuel purchase. Furthermore, Guha, Meyerson, and Munagala (2000) reported another application of MQC, namely load balanced facility location for layered network design. In their example, a franchise must open stores to minimize the average distance from customer to store. Meanwhile, it must also guarantee a minimum number of customers to each store so the individual stores remain profitable. Another application of MQC is presented by Brown, Dell, and Newman (2004) for planning U.S. military procurement in which the lower limit on the quantity of weapon systems available for purchase must be considered.

Although the MQC is widely applied in industries, study of its mathematical programming model and heuristics is rather scarce. However, other difficult restrictions have been studied for the transportation problem in the body of literature. For instance, the carrier's fixed selection cost was introduced by the facility location problem (FLP) (Hochbaum 1982; Cornuéjols, Nemhauser, and Wolsey 1990). Similar to the MQC, this restriction also makes the transportation problem intractable, i.e., *NP*-hard in the strong sense. Because of its practical importance, extensive research has been done on the FLP. Shmoys, Tardos, and Aardal (1997); Jain, Mahdian, and Saberi (2002); and Mahdian, Ye, and Zhang (2002) devised several constant-factor approximation algorithms, where the current best factor is 1.52. Because the FLP can also be formulated as an MIP model, its polyhedral structure has been well analyzed and some strong inequalities (or facets) have been discovered (Aardal and van Hoesel 1995b). Moreover, many heuristic algorithms have been applied to FLP and its variants, especially some advanced local search techniques, such as the genetic algorithms (Radcliffe 1993; Bozkaya, Zhang, and Erkut 2002), tabu search (Rolland, Schilling, and Current 1996; Al-Sultan and Al-Fawzan 1999).

Among works on different variants of FLP, Guha, Meyerson, and Munagala (2000) and Karger and Minkoff (2000) have studied the FLP with lower bound constraint. Guha, Meyerson, and Munagala

(2000) and Karger and Minkoff (2000) called it r-gathering problem. The lower bound constraint is the same as the MQC. They did not study the mathematical programming model of the problem, but devised constant-factor approximation algorithms for the metric version. In addition, approximative solutions proposed by their methods are not exactly feasible, but are allowed to break the lower bound constraint, i.e., satisfying only half of the minimum quantity for allocations. In this paper, we propose a greedy approximation algorithm whose approximation factor is not constant for the metric version, but its approximative solution strictly satisfies the minimum quantity constraint.

3. The Integer Programming Model

Before presenting the model, let us first show the complexity of the transportation problem with the MQC constraint. The following theorem claims that the new MQC constraint causes the transportation problem to be intractable even for approximations.

Theorem 1. *Whenever the minimum quantity, $b \geq 3$, the transportation problem with MQC is NP-hard in the strong sense and has no polynomial algorithm to guarantee a finite approximation factor unless $P = NP$.*

Proof. See Appendix A.

Furthermore, even if the transportation cost forms a metric, the problem is still NP-hard in the strong sense.

Theorem 2. *If the bid cost is metric and the minimum quantity $b \geq 3$, the transportation problem with MQC is NP-hard in the strong sense.*

Proof. See Appendix B.

In the rest of this section, we are going to provide an MIP model defined by a number of strong facets.

3.1. The Mixed-Integer Programming Model

We use a binary decision variable x_i for each carrier, i, and assign $x_i = 1$ if the carrier i has cargos to deliver, and $x_i = 0$ otherwise. Hence, it is straightforward to have the following integer programming (IP) model for the problem.

$$IP = \min \sum_{i \in I} \sum_{j \in J} c_{i,j} z_{i,j} \tag{4}$$

$$\text{s.t.} \sum_{i \in I} z_{i,j} = d_j, \quad \text{for } j \in J \tag{5}$$

$$b x_i \leq \sum_{j \in J} z_{i,j} \leq D x_i, \quad \text{for } i \in I \tag{6}$$

$$x_i \in \{0, 1\}, \quad \text{for } i \in I \tag{7}$$

$$z_{i,j} \in \mathcal{Z}_+, \quad \text{for } i \in I \text{ and } j \in J, \tag{8}$$

where $\mathcal{Z}_+$ is the set of nonnegative integers and $D = \sum_{j \in J} d_j$ is the total demands of customers. It can be seen that (4) minimizes the total transportation costs, restricted to the demand constraints (5) and the MQC constraint (6).

According to the IP model, we know that if values of the m binary variables, x_i, have been determined, the IP model becomes a new IP model denoted by $IP(S)$, where S represents the set of carriers with $x_i = 1$. The new model $IP(S)$ is further equivalent to a minimum cost-network flow problem with lower bounds on arcs (Ahuja, Magnanti, and Orlin 1993), which can be solved polynomially. Therefore, the optimum value of IP is determined by values of binary variables, x_i. Moreover, because the minimum quantity, b, and the demands, d_j, for $j \in J$ are integers, the network flow problem, $IP(S)$, will keep its convex hull unchanged if we relax integral $z_{i,j}$ to be continuous. Based on this, we can see that such an integral property is still held for $z_{i,j}$ in the IP model, which establishes the following theorem.

Theorem 3. *In the IP model, the convex hull of the constraints* (5) *through* (7) *and $z_{i,j} \in R_+$ (for $i \in I$ and $j \in J$) is the same as the convex hull of the IP, where R_+ indicates the set of nonnegative real numbers.*

According to Theorem 3, we can relax the integral variables $z_{i,j}$ and obtain the MIP model for the basic problem as follows:

$$MIP = \min \sum_{i \in I} \sum_{j \in J} c_{i,j} z_{i,j} \tag{9}$$

$$\text{s.t.} \sum_{i \in I} z_{i,j} = d_j, \quad \text{for } j \in J \tag{10}$$

$$b x_i \leq \sum_{j \in J} z_{i,j}, \quad \text{for } i \in I \tag{11}$$

$$\sum_{j \in J} z_{i,j} \leq D x_i, \quad \text{for } i \in I \tag{12}$$

$$x_i \in \{0, 1\}, \quad \text{for } i \in I \tag{13}$$

$$z_{i,j} \geq 0, \quad \text{for } i \in I \text{ and } j \in J. \tag{14}$$

In the MIP model, we separate the MQC constraint into two inequalities, (11) and (12). Inequality (12) will be improved in §3.2. Compared with the IP model that has $n + mn$ integral variables, the MIP has only m variables and is expected to be easier to solve. Moreover, although a feasible solution to the MIP model might have fractional assignments, $z_{i,j}$, for $i \in I$ and $j \in J$, the values of x_i are integral for $i \in I$. Based on the determined x_i, we can solve a corresponding $IP(S)$ model to get an integral feasible solution of the IP model with even better objective value.

3.2. Strengthening the Model

To improve the MIP model, let us first examine the strength of the inequalities in the current MIP model:

Theorem 4. *If $2b < D$, for the convex hull of the MIP model's feasible set,*

1. *its dimension is $mn + m - n$, and*
2. *the inequalities* (11), $x_i \le 1$ *for* $i \in I$, (14), *and equality* (10) *are facet-defining.*

Proof. Suppose $2b < D$ is satisfied. Here, we present the proof of the facet only for the most difficult case (11). For the other three inequalities, we can follow similar arguments (see Lim, Wang, and Xu 2003). Based on the following proof, the dimension of the convex hull can also be derived.

For any carrier $p \in I$, we are now going to show the inequality (11) for the carrier p, i.e.,

$$bx_p \le \sum_{j\in J} z_{p,j}, \tag{15}$$

is a facet. To see this, we must prove that if all feasible solutions of the *MIP* model that satisfy (15) at equality also satisfy

$$\sum_{i\in I}\sum_{j\in J}\alpha_{i,j} z_{i,j} + \sum_{i\in I}\beta_i x_i \le \theta \tag{16}$$

at equality, then (16) must be a linear combination of (15) and the equality constraint (10), which implies the inequality (15) is necessary, or facet-defining, for the *MIP* model.

First, let u and v denote any two different carriers in I other than p. We now show $\alpha_{u,j} = \alpha_{v,j}$ for all $j \in J$. Let us construct a feasible solution (x^1, z^1) as follows. Let $x_u^1 = x_v^1 = 1$ and other $x_i^1 = 0$ for $i \in I - \{u, v\}$. Because $2b < D$, there exists a feasible assignment $z_{i,j}^1$ for $i \in I$ and $j \in J$, such that $z_{u,j}^1 > 0$ and $z_{v,j}^1 > 0$ for all $j \in J$, and that $\sum_{j\in J} z_{u,j}^1 > b$ and $\sum_{j\in J} z_{v,j}^1 > b$ as well. Because $x_p = 0$, we know (x^1, z^1) satisfies (15) at equality. Consider another feasible solution (x^2, z^2), where $x_i^2 = x_i^1$ for all $i \in I$, and $z_{i,j}^2 = z_{i,j}^1$ for all $i \in I - \{u, v\}$ and $j \in J$, but

$$z_{u,j}^2 = z_{u,j}^1 + \epsilon \quad \text{and} \quad z_{v,j}^2 = z_{v,j}^1 - \epsilon, \qquad \text{for } j \in J,$$

where $\epsilon > 0$ is an arbitrary number that is close to 0. By appropriately choosing a small ϵ, we can keep (x^2, z^2) feasible and satisfying (15) at equality. By substituting (x^1, z^1) and (x^2, z^2) into (16) and subtracting 1 from the other, it leads to $\alpha_{u,j} = \alpha_{v,j}$. So, we can assume $\alpha_{u,j} = \alpha_j$ for each carrier $u \in I - \{p\}$, and rewrite (16) as

$$\sum_{j\in J}\alpha_j \sum_{i\in I} z_{i,j} + \sum_{j\in J}(\alpha_{p,j} - \alpha_j) z_{p,j} + \sum_{i\in I}\beta_i x_i \le \theta. \tag{17}$$

Second, we will show $\beta_u = 0$ for any $u \in I - \{p\}$. We construct another feasible solution (x^3, z^3) from (x^1, z^1) as follows. Let $x_i^3 = x_i^1$ for all $i \in I - \{u\}$, but $x_u^3 = 0$. Because only $x_v^3 = 1$, we assign $z_{v,j} = d_j$ for all $j \in J$. This leads (x^1, z^1) to be feasible and satisfy (15) at equality. Note

$$\sum_{j\in J}\alpha_j \sum_{i\in I} z_{i,j} = \sum_{j\in J}\alpha_j d_j \tag{18}$$

and $z_{p,j} = 0$ for $j \in J$. By substituting (x^1, z^1) and (x^3, z^3) into (17) at equality and subtracting 1 from the other, it leads to $\beta_u = 0$, for $u \in I - \{p\}$. We can rewrite (17) as

$$\sum_{j\in J}\alpha_j \sum_{i\in I} z_{i,j} + \sum_{j\in J}(\alpha_{p,j} - \alpha_j) z_{p,j} + \beta_p x_p \le \theta. \tag{19}$$

Third, we will prove $(\alpha_{p,g} - \alpha_g) = (\alpha_{p,h} - \alpha_h)$, for any two different requests $g, h \in J$. We construct a new feasible solution (x^4, z^4) by setting $x_p^4 = x_u^4 = 1$ and $x_i^4 = 0$ for $i \in I - \{p, u\}$. Because $2b < D$, there exists a feasible assignment $z_{i,j}^4$ for $i \in I$ and $j \in J$, such that $z_{p,j}^4 > 0$ and $z_{u,j}^4 > 0$ for $j \in J$, and that (15) is satisfied at equality. Consider another feasible solution (x^5, z^5), where $x_i^5 = x_i^4$ for $i \in I$, and $z_{i,j}^5 = z_{i,j}^4$ for $i \in I - \{p, u\}$ or $j \in J - \{g, h\}$, but

$$z_{p,g}^5 = z_{p,g}^4 + \epsilon, \qquad z_{p,h}^5 = z_{p,h}^4 - \epsilon,$$
$$z_{u,g}^5 = z_{u,g}^4 - \epsilon, \qquad z_{u,h}^5 = z_{u,h}^4 + \epsilon.$$

Hence, (x^5, z^5) keeps feasible and satisfies (15) at equality. Again, note the equality (18). By substituting (x^4, z^4) and (x^5, z^5) into (19) at equality and subtracting 1 from the other, it leads to $(\alpha_{p,g} - \alpha_g) = (\alpha_{p,h} - \alpha_h)$. Assuming $(\alpha_{p,g} - \alpha_g) = \lambda_p$ for each $g \in J$, we can rewrite (19) as

$$\sum_{j\in J}\alpha_j \sum_{i\in I} z_{i,j} + \lambda_p \sum_{j\in J} z_{p,j} + \beta_p x_p \le \theta. \tag{20}$$

Finally, because of (18) and $\sum_{j\in J} z_{p,j}^4 = b$, substituting (x^1, z^1) and (x^4, z^4) into (20) at equality and subtracting 1 from the other result in $\beta_p = -\lambda_p b$. For the same reason, by substituting (x^1, z^1) into (20) only, we have $\theta = \sum_{j\in J}\alpha_j d_j$. Therefore, we can rewrite (20) as

$$\sum_{j\in J}\alpha_j \sum_{i\in I} z_{i,j} + \lambda_p \sum_{j\in J} z_{p,j} \le \sum_{j\in J}\alpha_j d_j + \lambda_p b x_p, \tag{21}$$

which is a linear combination of (15) and the equality constraint (10). This concludes the proof of the facet of (11).

Let us now consider the dimension of the convex hull defined by the *MIP* model. In (16), there are $mn + m + 1$ unknown parameters $\alpha_{i,j}$, β_i, and θ for $i \in I$ and $j \in J$. Along with the *MIP* model and (15) at equality, we solve these equations in (16) and obtain (21). Because only $n + 1$ unknown ones are left in (21), noting that n independent equations are in (10), we know that the maximum number of affinely independent feasible solutions in the facet is exactly $mn+m-n$. This implies the dimension of the facet is $mn+m-n-1$, which is 1 less than that of the whole convex hull. So, the dimension of the convex hull is $mn+m-n$. □

To strengthen the model, we reformulate (12) as

$$z_{i,j} \le d_j x_i, \quad \text{for } i \in I \text{ and } j \in J, \tag{22}$$

which is a family of facets of the *MIP* model if $2b < D$. It must be noticed that using (22) to replace (12) is a standard way to strengthen the model of the FLP (Balinski 1966). In our experiments in §6, we will show that this replacement for FLP is also effective for our proposed transportation problem with MQC. We are now going to show that the inequalities in (22) form a facet if $2b < D$.

Theorem 5. *If $2b < D$, the inequalities in* (22) *are facet-defining for the convex hull of the MIP model's feasible set.*

Proof. Suppose $2b < D$ is satisfied. For any carrier $p \in I$ and any request $q \in J$, we are going to show that the inequality $z_{p,q} \le d_q x_p$ is a facet. To see this, we must prove that if all feasible solutions of the *MIP* model that satisfy $z_{p,q} = d_q x_p$ also satisfy (16) at equality, then (16) must be a linear combination of $z_{p,q} = d_q x_p$ and the equalities (10).

Following the same arguments as the first and second steps of the proof for the case (11) in Theorem 4, we can obtain $\alpha_{u,j} = \alpha_{v,j} = \alpha_j$ and $\beta_u = 0$ for $u \in I-\{p\}$, $v \in I-\{p,u\}$, and $j \in J$. So, (16) can be rewritten as (19).

Now we are going to show $(\alpha_{p,g} - \alpha_g) = 0$ for any request g in J but other than q. Similar to the proof in Theorem 4, we construct a new feasible solution $(\tilde{x}^4, \tilde{z}^4)$ as follows. We still let $\tilde{x}^4_p = \tilde{x}^4_u = 1$ and $\tilde{x}^4_i = 0$ for $i \in I-\{p,u\}$. For the feasible assignments, we still require $\tilde{z}^4_{p,j} > 0$ and $\tilde{z}^4_{u,j} > 0$ for $j \in J-\{q\}$, but $\tilde{z}^4_{p,q} = d_q$ and $\tilde{z}^4_{u,q} = 0$ in addition. Because $2b < D$, such a feasible assignment must exist. Now let us consider another feasible solution $(\tilde{x}^5, \tilde{z}^5)$, where $\tilde{x}^5_i = \tilde{x}^4_i$ for $i \in I$, and $\tilde{z}^5_{i,j} = \tilde{z}^4_{i,j}$ for $i \in I-\{p,u\}$ or $j \in J-\{g\}$, but we let

$$\tilde{z}^5_{p,g} = \tilde{z}^4_{p,g} + \epsilon, \qquad \tilde{z}^5_{u,g} = \tilde{z}^4_{u,g} - \epsilon.$$

By choosing a sufficiently small positive ϵ, we can keep $(\tilde{x}^5, \tilde{z}^5)$ feasible and satisfying $z_{p,q} = d_q$. By (18), substituting $(\tilde{x}^4, \tilde{z}^4)$ and $(\tilde{x}^5, \tilde{z}^5)$ into (19) at equality and subtracting one from the other results in $(\alpha_{p,g} - \alpha_g) = 0$ for any $g \in J-\{q\}$. Therefore, we can rewrite (19) as

$$\sum_{j\in J} \alpha_j \sum_{i\in I} z_{i,j} + (\alpha_{p,q} - \alpha_q) z_{p,q} + \beta_p x_p \le \theta. \tag{23}$$

Finally, recall that (x^1, z^1) is a feasible solution constructed in the proof of Theorem 4, and $x^1_p = 0$ and $z^1_{p,j} = 0$ for $j \in J$. Because of (18) and $\tilde{z}^4_{p,q} = d_j \tilde{x}^4_p$, substituting (x^1, z^1) and $(\tilde{x}^4, \tilde{z}^4)$ into (20) at equality and subtracting one from the other result in $\beta_p = (\alpha_q - \alpha_{p,q}) d_q$. For the same reason, by substituting (x^1, z^1) into (20) only, we have $\theta = \sum_{j\in J} \alpha_j d_j$. Therefore, we can rewrite (20) as

$$\sum_{j\in J} \alpha_j \sum_{i\in I} z_{i,j} + (\alpha_{p,q} - \alpha_q) z_{p,q} \le \sum_{j\in J} \alpha_j d_j + (\alpha_{p,q} - \alpha_q) d_q x_p, \tag{24}$$

which is a linear combination of $z_{p,q} = d_q z_p$ and the equality constraint (10). □

Furthermore, the old inequalities in (12) are redundant because they can be derived by summing up some new inequalities in (22). We can therefore replace (12) with (22) in the *MIP* and have the following SMIP model:

$$SMIP = \min \sum_{i\in I} \sum_{j\in J} c_{i,j} z_{i,j} \tag{25}$$

$$\text{s.t.} \quad \sum_{i\in I} z_{i,j} = d_j, \quad \text{for } j \in J \tag{26}$$

$$b x_i \le \sum_{j\in J} z_{i,j}, \quad \text{for } i \in I \tag{27}$$

$$z_{i,j} \le d_j x_i, \quad \text{for } i \in I \text{ and } j \in J \tag{28}$$

$$x_i \in \{0,1\}, \quad \text{for } i \in I \tag{29}$$

$$z_{i,j} \ge 0, \quad \text{for } i \in I \text{ and } j \in J. \tag{30}$$

Before we end this section, let us make some comments on the condition $2b < D$ of the facets. Intuitively speaking, if the minimum quantity, b, is close to the total demand, D, only one or two carriers can be selected to have $x_i = 1$ at the same time, and the problem will become very simple. For instance, if $2b > D$, at most one carrier can have cargos, and only m feasible solutions exist to be explored. Therefore, except for a few simple instances, the condition $2b < D$ appears trivial.

4. Branch and Cut

In *SMIP*, there are m binary variables, x_i, for $i \in I$ to be determined. Remember that if $\{x_i\}$ are determined, we can let $S = \{i \in I: x_i = 1\}$ and solve the $IP(S)$ model to generate its optimum assignment, $z_{i,j}$, for $i \in I$ and $j \in J$. Therefore, simply enumerating all the possible values of $\{x_i\}$ is enough to find an optimum solution for the *SMIP* model. However, such a simple exhaustive search scheme cannot work in practice because of the exponential search space.

To improve the efficiency, we need to prune invalid branches during the search process. This leads to the following branch-and-bound method. Let (x^*, z^*) denote the current best feasible solution. Suppose that we have assigned values to some x_i, and accordingly, let Π_0 (or Π_1) indicate the set of x_i determined to be 0 (or 1). We can use a pair (Π_0, Π_1) to represent a partially determined $\{x_i\}$. Before exploring the undetermined x_i further, let us make a lower bound estimation of the objective cost. Unless the lower bound is less than the current minimum cost of (x^*, z^*), we do not need to assign the values to the remaining x_i further for $i \in I - (\Pi_0 \cup \Pi_1)$. To make such a lower bound, we keep those values of determined x_i for $i \in \Pi_0 \cup \Pi_1$ and relax the rest of the undetermined x_i from binary to the closed interval $[0, 1]$. This changes the *SMIP* model to a relaxed linear programming (LP) model:

$$LP(\Pi_0, \Pi_1)$$

$$= \min \sum_{i\in I}\sum_{j\in J} c_{i,j} z_{i,j} \tag{31}$$

$$\text{s.t.} \sum_{i\in I} z_{i,j} = d_j, \quad \text{for } j \in J \tag{32}$$

$$b x_i \le \sum_{j\in J} z_{i,j}, \quad \text{for } i \in I \tag{33}$$

$$z_{i,j} \le d_j x_i, \quad \text{for } i \in I \text{ and } j \in J \tag{34}$$

$$x_i = 0, \quad \text{for } i \in \Pi_0 \tag{35}$$

$$x_i = 1, \quad \text{for } i \in \Pi_1 \tag{36}$$

$$x_i \in [0, 1], \quad \text{for } i \in I - (\Pi_0 \cup \Pi_1) \tag{37}$$

$$z_{i,j} \ge 0, \quad \text{for } i \in I \text{ and } j \in J, \tag{38}$$

whose optimum solution, denoted by (x', z'), gives a lower bound of the further exploration. If the cost of (x', z') is not less than the cost of the current best solution (x^*, z^*), no better solution will appear in further exploration for (Π_0, Π_1). We therefore stop exploring the rest of the undetermined x_i for $i \in I - (\Pi_0 \cup \Pi_1)$ and turn to other unexplored pairs (Π'_0, Π'_1). Otherwise, the cost of (x', z') is better. Let us then try different values for an undetermined x_i, where $i \in I - (\Pi_0 \cup \Pi_1)$, and generate new pairs for further exploration. This pruning technique reduces the size of the space to be explored and then improves the performance of the search scheme.

To further enhance the tightness, we adopt the branch-and-cut algorithm (Wolsey 1998, Aardal and van Hoesel 1995a). This algorithm is based on the branch-and-bound method, but adds new constraints to strengthen the model during the search process. Remember that during the process of the branch and bound, we have obtained a solution (x', z') for the relaxed LP model $LP(\Pi_0, \Pi_1)$. If x' is integral, then (x', z') is a feasible solution of the *SMIP* model and must be exactly the best solution that can be found in further exploration. If x' is fractional, the solution (x', z') is unfeasible for the *SMIP* model. In this case, an intuitive way to strengthen the *SMIP* model is to add a constraint, or cutting plane, with which all the feasible solutions are satisfied except for (x', z'). The unfeasible (x', z') is now excluded from the relaxed LP model and the MIP model becomes tighter.

5. Heuristics

5.1. A Linear Programming Rounding Heuristic Recalling that the problem's optimum value is determined by values of x_i where $i \in I$, we can iteratively choose some x_i and set it to 1 through an LP rounding heuristic, which is shown in Algorithm 1.

During each iteration, let (x^*, z^*) denote the current best feasible solution found for the *SMIP* model. Let Π_1 denote the set of x_i currently determined to be 1. Then, for those undetermined carriers, $k \in I - \Pi_1$, we can relax x_k to be continuous in the interval $[0, 1]$. Keeping other inequalities in *SMIP* unchanged, we obtain a relaxed LP model, denoted by $LP(\Pi_1)$.

By solving $LP(\Pi_1)$, we can get its optimum fractional solution, (x', z'), which may not be feasible for the *SMIP* model. However, the total transportation cost of (x', z') gives a lower bound estimation for further exploration. If it is not less than the current minimum cost of (x^*, z^*), we stop the algorithm and return to (x^*, z^*) as the near-optimum solution. Otherwise, the undetermined x_i that has the largest fractional value x'_i will be chosen and rounded to 1. Therefore, we have a new relaxed $LP(\Pi_1 \cup \{i\})$ model for the next iteration. By solving the $IP(S)$ where $S = \Pi_1 \cup \{i\}$, we can have the best assignment, $z_{i,j}$, under the determined $x_i \in S$ and obtain a feasible solution $(\tilde{x}, \tilde{z})$. If its cost is better than that of the current best, (x^*, z^*), we should replace (x^*, z^*) with $(\tilde{x}, \tilde{z})$.

Obviously, the solution returned by Algorithm 1 is feasible for the problem and its distance to the optimum depends on the accuracy of the lower bound estimation by $LP(\Pi_1)$. The reason for this is that if the lower bound estimation is closer to the exactly optimum value, the fractional solution will be closer to the integral one, and the cost increased by rounding the fractional to the integral will be smaller. This stresses the importance of our efforts on strengthening the MIP model in §3.2. Moreover, the experiments in §6 show that our LP rounding heuristic performs accurately and efficiently in practice.

Algorithm 1: LP Rounding Heuristic.

1: Let (x^*, z^*) be the current best feasible solution. Initially, (x^*, z^*) is empty with infinite cost.
2: Let Π_1 denote the set of x_i determined to be one. Initially, $\Pi_1 \leftarrow \varnothing$.
3: **while** $\Pi_1 \ne I$ **do**
4: Solve the relaxed linear programming model, $LP(\Pi_1)$, and obtain its optimum fractional solution, (x', z');
5: **if** the cost of (x', z') is NOT less than that of the current best (x^*, z^*) **then**

```
6:        There is no need to explore further. Stop the
          iteration and goto 18;
7:     else
8:        Select an x'_k whose fractional value is largest
          among x'_i for i ∈ I − Π_1;
9:        Round x'_k to one by S ← Π_1 ∪ {k};
10:       Solve the IP(S) model, obtaining its best
          feasible assignments, z̃;
11:       Set x̃_i ← 1 for i ∈ S, and x̃_i ← 0 otherwise;
12:       if the cost of (x̃, z̃) is less than that of the
          current best (x*, z*) then
13:          Replace (x*, z*) ← (x̃, z̃).
14:       end if
15:       Π_1 ← S, we have the new relaxed LP(Π_1)
          model for the next iteration;
16:    end if
17: end while
18: Return (x*, z*) as the near-optimum.
```

5.2. GREEDY Approximation Heuristics

Let us assume that the demand, d_j, of each customer $j \in J$ is 1, because if $d_j > 1$ for some $j \in J$, we can split the customer j to d_j customers, each having unit demand. Thus, the customer set, J, can represent the cargos set. Let C denote the set of cargos that have not been delivered yet, and Π_1 denote the set of carriers who have been selected to deliver some cargos, i.e., $x_i = 1$. The idea of our greedy method is as follows.

First, we define the following two elemental operations:

(1) For each unselected carrier, $i \in I - \Pi_1$, we define an operation, *selection*(i), which sets $x_i = 1$ to select carrier i for delivering some cargos and assigns it the b cheapest undelivered cargos in $J - C$ to satisfy the MQC.

(2) For each selected carrier $i \in \Pi_1$, we define an operation, *assignment*(i), which assigns carrier i the cheapest undelivered cargo in $(J - C)$ for delivering.

We measure each operation by its average operation cost. For *selection*(i), the average operation cost is $(\sum_{j \in A} c_{i,j})/b$, where A is the set of b cargos assigned to carrier i. For *assignment*(i), the average operation cost is $c_{i,j}$, where j is the cargo assigned to carrier i.

Based on the above, a feasible solution can be constructed by a sequence of elemental operations. In our greedy scheme shown in Algorithm 2, we do the operation iteratively that has the minimum average operation cost until all the cargos in J have been delivered. Appendix D shows that the time complexity of Algorithm 2 is $O(n^2 m)$ for the unit-demand case, and for the case when demands are not unit, the time complexity is $O(n^2 m + nm^2)$.

To analyze the solution quality produced by the greedy heuristic, let us consider the following theorem, which claims that Algorithm 2 can guarantee a $2b$ approximation factor when the transportation cost is a metric, and therefore if b is given as a constant, the factor is a constant as well.

Theorem 6. *When the transportation cost forms a metric, the greedy heuristic generates a feasible solution whose total transportation cost is at most* $2b$ *times the optimum.*

Proof. Under the assumption that the transportation cost $c_{i,j}$ forms a metric for $i \in I$ and $j \in J$, we are going to prove that the greedy Algorithm 2 generates a feasible solution whose total cost is at most $2b$ times the optimum. Because we have shown that any instance can be transformed to the unit-demand case by splitting, we can assume that $d_j = 1$ for all $j \in J$ without invalidating the approximation factor that we will prove. Therefore, the client set J represents the cargos set as well. When $b = 1$, because we assign each cargo to the carrier whose transportation cost is the lowest, the greedy Algorithm 2 generates the optimum solution and its approximation factor is certainly at most $2b$.

Consider the case when $b \geq 2$. For the optimum assignment, we let A_i denote the set of cargos assigned to i for each carrier $i \in I$, and n_i denote the size of A_i. By the MQC, we know either $n_i = 0$ or $n_i \geq b$. Let $\overline{opt}_i$ denote the cost of assignments to the carrier i, so that $\overline{opt}_i = \sum_{j \in A_i} c_{i,j}$. For the greedy assignment, let $\delta(j)$ denote the carrier to whom the cargo j is assigned for each cargo $j \in J$, and $f(j)$ represent the cost of the operation, *selection*$(\delta(j))$ or *assignment*$(\delta(j))$, which assigns j to $\delta(j)$ in the greedy Algorithm 2. Hence, the optimum total cost is $\sum_{i \in I} \overline{opt}_i$, and the greedy total cost is $\sum_{i \in I} \sum_{j \in A_i} f(j)$ by partitioning the cargos set J into $A_1, A_2, \ldots, A_m$. To show the approximation factor of 2 for the greedy algorithm, it is sufficient to prove that

$$\sum_{i \in I} \sum_{j \in A_i} f(j) < 2b \sum_{i \in I} \overline{opt}_i, \tag{39}$$

which can be directly drawn from the following Lemma 1 by summing up (40) for $i \in I$. (See Appendix C for the proof of Lemma 1.)

Lemma 1. *For each carrier* $i \in I$, *we have*

$$\sum_{j \in A_i} f(j) < 2b\, \overline{opt}_i. \quad \square \tag{40}$$

To see that $2b$ is close to the best possible approximation guarantee for the greedy Algorithm 2, consider the following instance in which the greedy cost is b times the optimum. Suppose the minimum quantity is b. We have two carriers to deliver $2b$ cargos. Carrier 1 delivers each of the first $b + 1$ cargos for free, but delivers each of the rest $b - 1$ ones with c cost. On the contrary, Carrier 2 can deliver each of the first $b + 1$ cargos with c cost, but deliver each of the rest $b - 1$ ones for free. Such an instance satisfies the metric condition. By applying the greedy heuristics to this instance, we select Carrier 1 to deliver the first $b+1$ cargos first, and assign the same Carrier 1 to deliver the rest $b-1$ ones as well, so that the total cost by greedy is $(b-1)c$. Because the optimum cost is obviously c, the approximation factor of the greedy heuristics is at least $b-1$ for the metric version.

Algorithm 2: GREEDY Heuristic.

1: Suppose all demands d_j are unit for $j \in J$, since, otherwise, if some $d_j > 1$, we can split the customer, j, into d_j customer each with unit demand.
2: Initially, both the selected carrier set, Π_1, and the delivered cargos set, C, are empty, and $x_i \leftarrow 1$ for $i \in I$;
3: **while** NOT all cargos have been delivered, i.e., $C \neq J$ **do**
4:    Choose an operation σ with minimum cost among all *selection*(i) for $i \in I - \Pi_1$ and *assignment*(i) for $i \in \Pi_1$;
5:    **if** σ is *selection*(i) **then**
6:       Select carrier i by $x_i \leftarrow 1$ and $\Pi_1 \leftarrow \Pi_1 \cup \{i\}$;
7:       Let A denote the set of b cargos whose transportation costs are the b cheapest $c_{i,j}$ for $j \in J - C$.
8:       For each cargo $j \in A$, assign it to carrier i for delivering, so $z_{i,j} \leftarrow 1$ and $C \leftarrow C \cup \{j\}$;
9:    **else if** σ is *assignment*(i) **then**
10:       Let j denote the undelivered cargo that minimizes the transportation cost $c_{i,j}$ for $j \in J - C$;
11:       Assign the cargo j to carrier i for delivering, so $z_{i,j} \leftarrow 1$ and $C \leftarrow C \cup \{j\}$;
12:    **end if**
13: **end while**
14: Return $\{x, z\}$ as the approximated solution.

6. Experiments

6.1. Generating Test Instances

There are two types of test instances. One is randomly generated, whose transportation cost, $c_{i,j}$, is randomly generated by a uniform distribution on the continuous interval $[0, 1]$. The other is metric based, whose $c_{i,j}$ is the distance between two of the $m+n$ random points in a planar rectangle $[0, 100] \times [0, 100]$. In the second type, the cost, $c_{i,j}$, forms a metric. For each type, we then have three kinds of instance size, (m, n). They were $(30, 60)$ for small size, $(60, 120)$ for medium size, and $(90, 120)$ for large size. In total, we have six groups of instances: R-I (or M-I) is random (or metric) small, R-II (or M-II) is random (or metric) medium, and R-III (or M-III) is random (or metric) large.

For each group, we generate 10 different sets of transportation costs, $c_{i,j}$, and demands, d_j. The cost $c_{i,j}$ are generated according to its type, and the demands d_j for $1 \le j \le m$ were uniform random numbers on the integral set $\{10, 11, \ldots, 100\}$.

Table 1 Configurations of Each Group of Test Cases

Group ID[1]	Number of agents m	Number of requests n	Transportation cost $c_{i,j}$
R-I	30	60	Random
R-II	60	120	Random
R-III	90	180	Random
M-I	30	60	Metric
M-II	60	120	Metric
M-III	90	180	Metric

[1]Character "R"/"M" represents that the type of transportation cost is random/metric, and its roman index indicate the instance size, i.e., I for small, II for medium, and III for large.

Finally, for each set of d_j and $c_{i,j}$, we generate 20 different minimum quantities by $b = \lfloor D/(mw_b) \rfloor$, where w_b is from $5\%, 10\%, \ldots,$ to 100%. This makes $\lfloor D/b \rfloor$ the maximum number of carriers having $x_i = 1$ appear uniform. As shown in Table 1, there are 200 test instances in each group.

We implemented all algorithms with Microsoft Visual C++ 6.0 and ran all the experiments on a Pentium III 800 MHZ PC with 128 M memory. By running the branch-and-cut search scheme provided by ILOG CPLEX 8.0 for an endurable long time, we obtained the best lower bound and the best upper bound of the optimum. Because the lower bound and upper bound were close to each other (as reflected in Table 2), they could be used as a good standard for comparisons of other experimental results.

6.2. Performance of the Models

We applied the branch-and-cut search schemes, constructed by the fundamental components from ILOG CPLEX 8.0, on both the *MIP* model and the *SMIP* model. Because the only difference between the two models was the facet (22), efficiency of the two search schemes reflected the performance of the models.

Table 2 Qualities of Best Lower Bound and Best Solution We Found

Group ID	Number of optimums proved[1]	Average difference between LB and BEST (%)[2]	Maximum difference between LB and BEST (%)[3]
R-I	200	0.00	0.00
R-II	199	0.02	3.30
R-III	180	0.04	13.11
M-I	200	0.00	0.00
M-II	200	0.00	0.00
M-III	199	0.01	1.58

[1]The number of optimum solutions found and proved among 200 instances within each group.
[2/3]Average/maximum difference between best lower bound (LB) and best solution value (BEST), as a percentage of best solution value, and over 200 instances within each group.

Table 3 Performance of the Branch-and-Cut Algorithm Tested Over the Small Problem Instances Within Groups R-I and M-I

	R-I		M-I	
	MIP	*SMIP*	*MIP*	*SMIP*
Average time (s)[1]	49.19	34.58	10.13	14.93
Maximum time (s)[2]	680.38	514.62	162.51	281.98
Number of optimums proved in 300s[3]	193	195	200	200

[1/2]Average/maximum time in seconds, consumed to find and prove an optimum solution, over 200 small problem instances within each test case group.

[3]The number of optimum solutions found and proved in 300 seconds and over 200 instances within each test case group.

We ran *SMIP* and *MIP* on both the small- and medium-scale instances. The experimental results shown in Table 3 show that both models handled the small-scale instances accurately, because they achieved the optimums in relatively short average times, and determined the optimums for most cases in 300 seconds.

However, for the medium-scale instances, *SMIP* outperformed *MIP*. Table 4 reveals the differences. Among the 200 instances of the medium group R-II (or M-II), *SMIP* achieved optimums for 152 (or 184) instances in 300 seconds, much more than *MIP*, achieved optimums for only 49 (or 38) instances in 300 seconds. By comparing the average difference between the lower bounds and the best solution values, we found that *SMIP* generated much closer lower bounds and solutions than *MIP* did and therefore converged much faster.

Additionally, we relaxed variables x_i to be continuous in *SMIP* and *MIP* to obtain the LP models denoted by *SLP* and *LP*, respectively. By solving the two relaxed models *SLP* and *LP* of each test instance within the six groups, we could obtain their lower bounds of optimums. From the results shown in Table 5, we observed that the lower bounds by *SLP* were much closer to the best solutions (on average 93.90%) than those by *LP* (on average 79.37%).

Table 4 Convergence Speed of the Branch-and-Cut Algorithm Over the Medium Problem Instances Within Groups R-II and M-II

	R-II		M-II	
	MIP	*SMIP*	*MIP*	*SMIP*
Number of optimums proved in 300s[1]	49	152	38	184
Average difference between LB and BEST in 300s[2]	18.21%	6.48%	6.54%	0.32%

[1]The number of optimum solutions, found and proved in 300 seconds, and over 200 medium problem instances within each test case group.

[2]Average difference between the best lower bound (LB) and the best solution value (BEST), found in 300 seconds, as a percentage of the latter and over 200 medium problem instances within each test case group.

Table 5 Average Difference Between the LP Lower Bound and the Best Solution Value

	Average difference from best solution value (%)[1]	
Group ID	*LP*	*SLP*
R-I	33.77	6.10
R-II	34.35	4.89
R-III	34.45	5.07
M-I	22.58	2.39
M-II	21.72	1.07
M-III	20.63	1.02

[1]Figures represent average difference between LP lower bound and best solution value, as a percentage of best solution value, and over 200 instances within each test case group.

Conclusively, the strengthened model *SMIP* outperformed the original model *MIP*. In other words, the new facet (22) is effective.

6.3. Performance of the Two Heuristics

Both the LP rounding heuristics (*LPR*) and the greedy-approximation heuristics (*GREEDY*) were tested by all instances within the six groups. For comparison, we also ran the *SMIP* search scheme, but with a 300-second running time limit. This search scheme is denoted by *SMIP*(300s). Its time limit is set to 300 seconds because the longest time consumed by the other two heuristics seldom exceeded 300 seconds.

Heuristic solutions by these three algorithms were compared with the best lower bounds we obtained. We measured the performance by comparing their gaps from the best lower bounds as a percentage of the best lower bounds:

$$GAP = \frac{SOL - LB}{LB} \times 100\%.$$

Their average gaps are shown in Table 6. Obviously, the *LPR* performed much better than the *GREEDY* and even better than the *SMIP* (300s) for some large size instances. Over all the cases, the maximum average gap of *LPR* was only 1.16%, which is much less than 6.85%, the minimum average gap of *GREEDY*.

Table 6 Average Gaps of the Heuristic Solutions from the Best Lower Bound

	Average gap from best lower bound (%)		
Group ID	*SMIP*(300s)	*LPR*	*GREEDY*
R-I	0.02	1.16	21.99
R-II	12.62	0.86	28.57
R-III	38.02	1.85	32.05
M-I	0.00	1.24	6.06
M-II	0.09	1.02	6.98
M-III	0.93	0.66	6.85

Moreover, the *LPR* performed well for both random groups (on average 1.29%) and metric groups (on average 0.97%). On the contrary, *GREEDY* generated worse solutions for random groups (on average 27.54%) than metric groups (on average 6.63%). In addition, over groups R-III and M-III of large size instances, the average gap of *SMIP*(300s) became worse, 38.02% for R-III and 0.93% for M-III, although, the average gap of *LPR* remained small, 1.85% for R-III and 0.66% for M-III. These observations demonstrate the good stability of the *LPR*.

In addition to the average gaps, we also counted the number of optimum solutions achieved by the heuristics. Table 7 summarizes the numbers and shows that *GREEDY* generated few optimums, but *LPR* generated several. It is interesting to see that the *LPR* achieved more optimums for random cases than metric ones, the opposite of *SMIP* (300s).

Furthermore, we examined the time consumed for each heuristic algorithm, which is presented in Table 8. *GREEDY* was no doubt the fastest heuristic, with an average time consumption of under one second for all the groups. The *LPR* was slower than *GREEDY*, but much faster than the *SMIP*(300s).

The *LPR* had good performance among all the groups, but we cannot prove its theoretical-performance guarantee. Remember that *GREEDY* has an approximation factor of $2b$, which makes us suspect that the solution generated by *GREEDY* may become better if b is decreasing. However, is this true in practice? And is this true for *LPR* as well?

To consider our suspicions, let us see how the two heuristics, *LPR* and *GREEDY*, perform in practice when b is decreasing. To make b decrease, we let w_b increase, because we have $b = \lfloor D/(mw_b) \rfloor$. Figure 1 shows the trends of the average gaps between the heuristic solutions and the best lower bounds when w_b is increasing. Within each of the two, large test groups, R-III and M-III, the average gaps become smaller for both *LPR* and *GREEDY*. The above observation has confirmed our suspicions on the effects of b. Moreover, we can also observe that *LPR* not only generated a better solution than *GREEDY*, but also had a stable behavior. To see this more clearly,

Table 7 **The Number of Optimums Found by the Heuristics**

	Number of optimums found		
Group ID	*SMIP* (300s)	*LPR*	*GREEDY*
R-I	199	131	0
R-II	167	86	0
R-III	124	47	0
M-I	200	96	5
M-II	190	56	0
M-III	167	30	0

Table 8 **Average Time Consumed by the Heuristics**

	Average time (sec)		
Group ID	*SMIP(300s)*	*LPR*	*GREEDY*
R-I	30.83	1.21	0.00
R-II	101.66	11.94	0.03
R-III	159.44	56.54	0.10
M-I	14.93	0.65	0.01
M-II	52.17	4.22	0.03
M-III	121.90	15.95	0.10

we present the statistics of Figure 1 in Table 9. Compared with *GREEDY*, *LPR* has a smaller average gap, a smaller standard deviation, and smaller absolute linear trends as well.

In summary, the LP rounding heuristics have good performance among all the various test instances in practice. However, the greedy approximation heuristic does not, except for its fast running speed.

7. Conclusion

To solve a transportation problem with the MQC constraint, in a practical way, we have studied its MIP model and strengthened the above model by adding a facet. In addition, we have applied two different heuristics to the problem. One is an LP rounding method, which performs well in experiments. The other is a combinatorial greedy approximation, which guarantees a nonconstant approximation factor for the metric version. We expect the study of MQC for

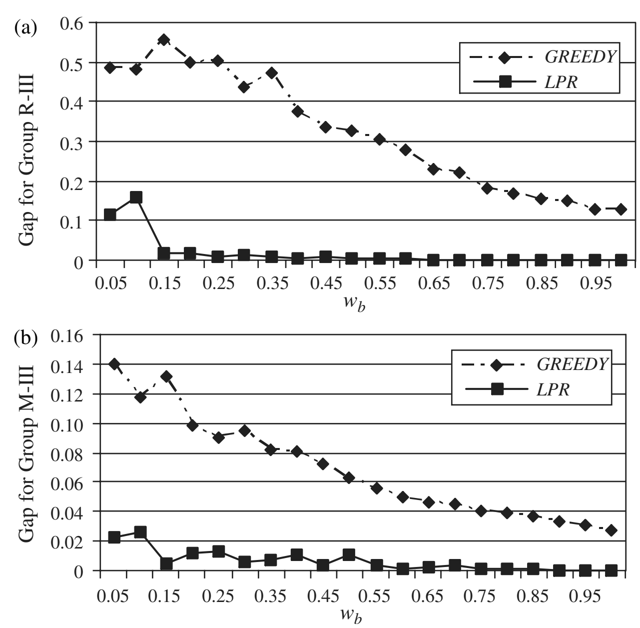

Figure 1 **Average Difference Between Heuristics Solution and Best Lower Bounds, Over 10 Large Problem Instances for Each w_b Increasing from 0.05 to 1.00, and Within Groups R-III and M-III, Respectively**

Table 9 Statistics of Heuristics Performance with w_b Increasing Over Large Problem Instances Within Groups R-III and M-III

	R-III		M-III	
	LPR	*GREEDY*	*LPR*	*GREEDY*
Average gap (%)[1]	1.85	32.05	0.66	6.85
Standard deviation[2]	0.04	0.15	0.01	0.03
Linear trend[3]	−0.08	−0.48	−0.02	−0.11

[1]Average difference between heuristics solution value and best lower bound, as percentage of best lower bound, and over 200 large problem instances within each test case group.

[2/3]Standard deviation/linear trend of the difference between heuristic solution value and best lower bound, over 200 large problem instances within each test case group.

the transportation problem to continue because of its scarceness in the literature and also its wide applications in industry.

Appendix A. Proof of the *NP*-Hardness and Inapproximation for the Transportation Problem with MQC

Proof. We only need to prove the case with $b = 3$, to which any case with $b > 3$ can be reduced. To prove its unary *NP*-hardness, we use a reduction from the following unary *NP*-complete problem.

Cover by 3-Sets (X3C). According to Garey and Johnson (1983) given a set $X = \{1, \dots, 3q\}$ and a collection $C = \{C_1, \dots, C_m\}$ with each member $C_i \subseteq X$ and $|C_i| = 3$ for $i = 1, \dots, m$, does C contain an exact cover for X, i.e., a subcollection $C' \subseteq C$ such that every element of X occurs in exactly one member of C'?

From any arbitrary instance of **X3C**, consider the following polynomial reduction to an instance of the transportation problem with MQC. Let each element $j \in X$ indicate a customer with a unit demand, then we have $J = \{1, \dots, 3q\}$ with $d_j = 1$ for $j \in J$. Suppose the carrier set I is $\{1, \dots, m\}$. For each carrier $i \in I$ and customer $j \in J$, the bid cost $c_{i,j} = 0$ if the element $j \in C_i$; otherwise, $c_{i,j} = 1$. Suppose the minimum quantity b is three units. Now we prove that its minimum total cost is 0 if and only if the **X3C** has an exact cover.

On one hand, if there exists an exact cover C' for **X3C**, we assign cargos to customers based on the exact cover C'. For each carrier $i \in I$ and customer $j \in J$, the assignment $z_{i,j} = 1$ if element j is covered by the subcollection C_i and C_i is in the exact cover C'; otherwise $z_{i,j} = 0$. Because C' is an exact cover, the demand constraint (2) is satisfied. Now we examine the MQC constraint (3). For each selected carrier $i \in A$, if $C_i \in C'$ then its total shipment to Region 1 is $\sum_{j\in J_1} z_{i,j} = |C_i| = 3$, otherwise $C_i \notin C'$ then we have $\sum_{j\in J_1} z_{i,j} = 0$. Both satisfy the MQC constraint, and its total cost (1) is 0, achieving the minimum.

On the other hand, if we have a feasible selection A and assignments $z_{i,j}$ for $i \in A$ and $j \in J$ with 0 total cost, consider the subcollection $C' = \{C_i \mid i \in A \text{ and } \sum_{j\in J} z_{i,j} > 0\}$. To see if C' is an exact cover of X, we need to prove that for each element $u \in X$ there exists a unique index p such that $C_p \in C'$ and $u \in C_p$. Because $X = J$, the corresponding customer $u \in J$. Noting $\sum_{i\in A} z_{i,u} = 1$ by the demand constraint (2), we can assume p is the unique index with $p \in A$ and $z_{p,u} = 1$, leading $\sum_{j\in J} z_{p,j} > 0$ and $C_p \in C'$. Because the total cost (1) is 0 and $z_{p,u} = 1$ implying $c_{p,u} = 0$, we have $u \in C_p$ that proves the existence. For the uniqueness, consider any other subset $C_q \in C'$ where $q \neq p$. We are going to prove the element $u \notin C_q$. Because for the customer u, its partial assignments $z_{q,u} + z_{p,u} \le 1$ and $z_{p,u} = 1$, we have $z_{q,u} = 0$. Note $C_q \in C'$ and $J_1 = J$, implying $\sum_{j\in J_1} z_{q,j} > 0$. To satisfy the MQC constraint (3), we have $\sum_{j\in J_1} z_{q,j} = 3$. This leads $\sum_{j\in C_q} z_{q,j} + \sum_{j\notin C_q} z_{q,j} = 3$. However, for $j \notin C_q$ the assignment $z_{q,j} = 0$, its bid cost $c_{q,j} = 1$, and the total cost (1) is 0. So, $\sum_{j\in C_q} z_{q,j} = 3$. For all $j \in C_q$, because $|C_q| = 3$ and its partial assignment $z_{q,j} \le 1$, we obtain $z_{q,j} = 1$. Noting $z_{q,u} = 0$ we have $u \notin C_q$, which leads the uniqueness and completes the proof of its unary *NP*-hardness.

Moreover, supposing $P \neq NP$, we can see that there exists no approximation algorithm with finite-approximation factor, otherwise, it can generate a feasible solution with 0 total cost in polynomial time if and only if the exact minimum total cost is 0. By the argument above, this condition is equivalent to the existence of an exact cover for the instance of **X3C**. So, **X3C** can be solved by the same algorithm polynomially. This implies **X3C** $\in P$, leading to a contradiction of our assumption of $P \neq NP$. □

Appendix B. Proof of the *NP*-Hardness for the Transportation Problem with MQC and Metric Transportation Costs

Proof. We can reduce any instance of the **X3C** problem to an instance of the transportation problem with MQC and metric bid costs through a similar way as we did in Appendix A, except that we set the bid costs in a different way. For each carrier i and j, the bid cost $c_{i,j}$ is set to be 1 if $j \in C_i$, and set to be 2 if $j \notin C_j$. The cost $c_{i,j}$, thus, forms a metric if we define the cost between any two different carriers or any two different customers is 1, while the cost from every carrier or every customer to itself is 0. Based on the same arguments in Appendix A, we can know that the instance of the **X3C** problem has a feasible exact cover if and only if the instance of the transportation problem with MQC has a feasible solution whose cost is no more than $3q$, noting that the element set in the **X3C** problem has a size of $3q$. □

Appendix C

Proof of Lemma 1. We prove (40) for the first carrier, i.e., $i = 1$ only, because for others, the proof is the same. We can also assume $n_1 \ge b$, otherwise A_1 is empty so that (40) is obviously true for $i = 1$.

Consider the n_1 cargos assigned to the Carrier 1 in the optimum assignments. Without loss of generality, they are supposed to be the cargos $1, 2, \dots, n_1$ that form the set A_1, and to be assigned in an nondecreasing order of time by the greedy algorithm. In other words, for any two cargos p and q where $1 \le p < q \le n_1$, the cargo p is assigned at least as early as q in the greedy Algorithm 2. If the carrier $\delta(p)$ has already been selected before we try to assign the cargo q greedily, the cost of *assignment*$(\delta(p))$ is at exactly $c_{\delta(p),q}$ at that time. Therefore, the operation cost $f(q)$ of assigning q must be at most $c_{\delta(p),q}$. Noting that $c_{i,j}$ forms a metric for $i \in I$ and $j \in J$, we have $c_{\delta(p),q} \le c_{\delta(p),p} + c_{1,p} + c_{1,q}$, and therefore,

$$f(q) \le c_{\delta(p),p} + c_{1,p} + c_{1,q}, \quad \text{for } 1 \le p < q \le n_1 \text{ and } \delta(p) \text{ is selected before assigning } q, \tag{41}$$

which gives an upper bound of the greedy operation cost.

To compute its lower bound, consider any cargo j where $1 \le j \le m$. Because either $f(j) = c_{\delta(j),j}$ if its greedy operation is *assignment*$(\delta(j))$, or $f(j) \ge c_{\delta(j),j}/b$ if that is *selection*$(\delta(j))$, we thus have

$$f(j) \ge \frac{c_{\delta(j),j}}{b}, \quad \text{for } 1 \le j \le n_1. \tag{42}$$

However, (42) and (41) are not enough. To obtain (40), we need a more sophisticated upper bound of the greedy operation for some cargos that have large transportation cost to the Carrier 1, say larger than or equal to $\overline{opt}_1/n_1$. Because $\sum_{j=1}^{n_1} c_{1,j} = \overline{opt}_1$, we know that $t = \min\{j \mid c_{1,j} \le \overline{opt}_1/n_1, j \in A_1\}$ must exist, which implies that

$$c_{1,j} \ge \frac{\overline{opt}_1}{n_1}, \quad \text{for } 1 \le j \le t-1. \tag{43}$$

Consider every cargo j where $1 \le j \le \min(t, n_1 - b + 1)$. We are now going to prove its greedy operation cost $f(j)$ is at most $\overline{opt}_1/n_1$. Before we assign the cargo j, cargo $1, 2, \ldots, j-1$ have been assigned. Because the cost of *selection*(1) or *assignment*(1) is not more than the $\sum_{p=j}^{n_1} c_{1,p}/(n_1 - j + 1)$ and the cost of $f(j)$ cannot exceed the cost of *selection*(1) or *assignment*(1), we thus have

$$f(j) \le \frac{\overline{opt}_1 - \sum_{k=1}^{j-1} c_{1,k}}{n_1 - j + 1}.$$

By (43), we have $\sum_{k=1}^{j-1} c_{1,k} \ge (j-1)\overline{opt}_1/n_1$; therefore, $f(j) \le \overline{opt}_1/n_1$, which implies that

$$f(j) \le \frac{\overline{opt}_1}{n_1}, \quad \text{for } 1 \le j \le \min(t, n_1 - b + 1). \tag{44}$$

Based on the above arguments, we can complete the proof of (40) for $i = 1$, by considering the following two cases.

For the first case, suppose $t \le n_1 - b + 1$, which implies $\min(t, n_1 - b + 1) = t$. Let B denote the set of b cargos assigned to $\delta(t)$ when $\delta(t)$ is first selected. Let $p = \max\{j\colon j \in B\}$ if $t \in B$, and let $p = t$ otherwise. By (44), we have

$$f(j) = f(t) \le \frac{\overline{opt}_1}{n_1}, \quad \text{for } t \le j \le p. \tag{45}$$

We write the greedy total cost in the following form:

$$\sum_{j \in A_1} f(j) = \sum_{j=1}^{p} f(j) + \sum_{j=p+1}^{n_1} f(j).$$

By (44) and (45), we know $f(j) \le \overline{opt}_1/n_1$ for $1 \le j \le p$. For $p+1 \le j \le n_1$ we have $f(j) \le (b+1)\overline{opt}_1/n_1 + c_{1,j}$ because $j \notin B$ and $f(j) \le c_{\delta(t),t} + c_{1,t} + c_{1,j}$ by (41), $c_{\delta(t),t} \le bf(t) \le b(\overline{opt}_1/n_1)$ by (42) and (44), and $c_{1,t} \le \overline{opt}_1/n_1$. Therefore, noting $b \ge 2$, $1 \le p$, and $\sum_{j=p+1}^{n_1} c_{1,j} \le \overline{opt}_1$, we have

$$\begin{aligned}\sum_{j \in A_1} f(j) &\le p(\overline{opt}_1/n_1) + (n_1 - p)(b+1)\overline{opt}_1/n_1 + \overline{opt}_1\\ &\le (b+2)\overline{opt}_1\\ &\le 2b\,\overline{opt}_1.\end{aligned}$$

For the second case, suppose $n_1 - b + 1 < t$, which implies $\min(t, n_1 - b + 1) = n_1 - b + 1$. Let B' denote the set of b cargos assigned to $\delta(1)$ when $\delta(1)$ is first selected. Let $p' = \max\{j\colon j \in B\}$ if $1 \in B'$, and let $p' = 1$ otherwise. By (44), we have

$$f(j) = f(1) \le \frac{\overline{opt}_1}{n_1}, \quad \text{for } t \le j \le p'. \tag{46}$$

Similar to the first case, we write the greedy total cost in a new form, i.e.,

$$\sum_{j \in A_1} f(j) = \sum_{j=1}^{\max(n_1-b+1,\,p')} f(j) + \sum_{j=\max(n_1-b+1,\,p')+1}^{n_1} f(j).$$

Let $K = \max(n_1 - b + 1, p')$. By (44) and (46), we get $f(j) \le \overline{opt}_1/n_1$ for $j \le K$. Hence, if $K = n_1$, then $\sum_{j \in A_1} f(j) \le \overline{opt} \le 2b\,\overline{opt}$. Now let us consider the case when $K < n_1$. For $K + 1 \le j \le n_1$ we have $f(j) \le b(\overline{opt}_1/n_1) + c_{1,1} + c_{1,j}$ because $j \notin B'$ so that $f(j) \le c_{\delta(1),1} + c_{1,1} + c_{1,j}$ by (41) and $c_{\delta(1),1} \le bf(1) \le b(\overline{opt}_1/n_1)$ by (42) and (44). Therefore, noting $\sum_{j=K+1}^{n_1} c_{1,j} \le \overline{opt}_1 - c_{1,1}$ and $b \le n_1$ and $c_{1,1} \le \overline{opt}_1$ and $n_1 - b + 1 \le K < n_1$, we have

$$\begin{aligned}\sum_{j \in A_1} f(j) &\le K\,\overline{opt}_1/n_1 + (n_1 - K)b(\overline{opt}_1/n_1)\\ &\quad + (n_1 - K)c_{1,1} + (\overline{opt}_1 - c_{1,1})\\ &\le [n_1^2 + nb - K(n_1 + b - 1)]\overline{opt}/n_1\\ &\le \left(b + \frac{(b-1)^2}{n_1}\right)\overline{opt}_1\\ &\le 2b\,\overline{opt}_1. \quad \square\end{aligned}$$

Appendix D. Analysis of Time Complexity for GREEDY Algorithm 2

Let us discuss the time complexity of the GREEDY Algorithm 2. First, we assume that demand d_j is one unit for every client $j \in J$. Therefore, the number of cargos to be assigned is n, implying that the number of iterations is at most n. In each iteration, exactly m operations will be considered. The cost of each operation can be computed in $O(b)$ times because we can list the cargo $j \in J$ in nondecreasing order of $c_{i,j}$ for each carrier $i \in I$, respectively, and then compute the operation cost in the following way. For each unselected carrier $i \in \Pi$, the first b unassigned cargos in its cargos list contribute to the cost of *selection*(i). On the other hand, for each selected carrier $i \in I - \Pi$, only the first unassigned cargo in its cargos list contributes to the cost of *assignment*(i). Moreover, because those assigned carriers will be deleted from the list when met again, there are n deletions total for each list. Amortizedly speaking, computing the cost of each operation takes at most $O(b + n/n)$, i.e., $O(b)$ time, on average. Because we may assume b is at most n

otherwise no feasible solution exists, the time complexity of Algorithm 2 is $O(nmb)$, or $O(n^2m)$, which is polynomial to the length of instance.

Now, let us consider the general case where d_j is not one unit for some $j \in J$. In this case, the number of cargos equals the total demand D, which leads the time complexity of the greedy Algorithm 2 to $O(Dmb)$. Because $O(Dmb)$ may be exponential to the length of the instance, we have to improve our implementation in a more careful manner to keep the polynomial-time complexity for this general case.

The improvement is based on the fact that the cargos of the same client $j \in J$ have the same transportation cost for all $i \in I$. Therefore, if *assignment*(i) is the current operation that assigns a cargo of client j to the carrier i, then the same operation will be kept going until all the rest of the cargos of j are assigned to i. For this reason, we can assign the remaining cargos of client j together to the carrier i in one iteration. In this way, at most m operations of *selection*(i) and n operations of *assignment*(i) will be done, so the number of iterations is at most $m+n$. For the same reason, to compute the cost of each operation, we only need to know the number of unassigned cargos left for every client in its list. Because the length of the list is at most n, the cost of each operation can be computed in $O(n)$ time. Because there are m operations considered for each iteration, the total time complexity is reduced to $O((n+m)mn)$, or $O(n^2m+nm^2)$, which remains polynomial for the general case.

Therefore, the greedy Algorithm 2 can be implemented in a careful way so that its time complexity is polynomial to the length of instance: $O(n^2m)$ for the unit-demand case and $O(n^2m+nm^2)$ for the general case.

References

Aardal, K., C. van Hoesel. 1995a. Polyhedral techniques in combinatorial optimization: Part I, theory. *Statistica Neerlandica* **50**(3).

Aardal, K., C. van Hoesel. 1995b. Polyhedral techniques in combinatorial optimization: Part II, applications and computations. *Statistica Neerlandica* **50**(3).

Ahuja, R. K., T. L. Magnanti, J. B. Orlin. 1993. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, Englewood Cliffs, NJ.

Al-Sultan, K., M. Al-Fawzan. 1999. A tabu search approach to the uncapacitated facility location problem. *Ann. Oper. Res.* **86** 91–103.

Balinski, M. 1966. On finding integer solutions to linear programs. *Proc. IBM Sci. Computing Symp. on Combin. Problem*, IBM, 225–248.

Bassok, Y., R. Anupindi. 1997. Analysis of supply contracts with total minimum commitment. *IIE Trans.* **29**(5).

Bonser, J., S. Wu. 2001. Procurement planning to maintain both short-term adaptiveness and long-term perspective. *Management Sci.* **47**(6) 769–786.

Bozkaya, B., J. Zhang, E. Erkut. 2002. A genetic algorithm for the p median problem. Z. Drezner, H. Hamacher, eds. *Facility Location: Applications and Theory*. Springer, Berlin, Germany.

Brown, G., R. Dell, A. Newman. 2004. Optimizing military capital planning. *Interfaces* **34**(6) 415–425.

Chen, F., D. Krass. 1999. Analysis of supply contracts with minimum total order quantity commitments and nonstationary demands. *Eur. J. Oper. Res.* **131** 309–323.

Cornuéjols, G., G. L. Nemhauser, L. A. Wolsey. 1990. The uncapacitated facility location problem. P. Mirchandani, R. Francis, eds. *Discrete Location Theory*. Wiley, New York, 119–171.

Garey, M. R., D. S. Johnson. 1983. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, New York.

Guha, S., A. Meyerson, K. Munagala. 2000. Hierarchical placement and network design problems. *Proc. 41st Annual IEEE Sympos. Foundations Comput. Sci.*, IEEE Computer Society, Washington, D.C., 603–612.

Hochbaum, D. S. 1982. Heuristics for the fixed cost median problem. *Math. Programming* **22** 148–162.

Jain, K., M. Mahdian, A. Saberi. 2002. A new greedy approach for facility location problems. *Proc. 34th ACM Sympos. Theory Comput. (STOC)*, ACM Press, New York.

Karger, D. R., M. Minkoff. 2000. Building Steiner trees with incomplete global knowledge. *Proc. 41st Annual IEEE Sympos. Foundations Comput. Sci.*, IEEE Computer Society, Washington, D.C., 613–623.

Lim, A., F. Wang, Z. Xu. 2003. A transportation problem with minimum quantity commitments. Working paper, Hong Kong University of Science and Technology, Hong Kong.

Mahdian, M., Y. Ye, J. Zhang. 2002. Improved approximation algorithms for metric facility location problems. *Proc. 5th Int. Workshop Approximation Algorithms Combin. Optim.*, Springer-Verlag, London, U.K., 229–242.

Radcliffe, N. J. 1993. Genetic set recombination. L. D. Whitley, ed. *Proc. 2nd Workshop Foundations Genetic Algorithms*. Morgan Kaufmann Publishers, San Mateo, CA, 203–219.

Rolland, E., D. A. Schilling, J. R. Current. 1996. An efficient tabu search procedure for the p-median problem. *Eur. J. Oper. Res.* **21** 329–342.

Shmoys, D. B., E. Tardos, K. Aardal. 1997. Approximation algorithms for facility location problems (extended abstract). *Proc. 29th Annual ACM Sympos. Theory Comput.*, ACM Press, New York, 265–274.

Wolsey, L. A. 1998. *Integer Programming*. John Wiley & Sons, Inc., New York.