DISCUSSION PAPER NO. 98

TRANSPORTATION TYPE PROBLEMS WITH QUANTITY DISCOUNTS

V. Balachandran* and A. Perry*

July 31, 1974

*Graduate School of Management
Northwestern University
Evanston, IL 60201
ABSTRACT

It is known, to be real, that the per unit transportation cost from a specific supply source to a given demand sink is dependent on the quantity shipped, so that there exists finite intervals for quantities where price breaks are offered to customers. Thus, such a quantity discount results in non convex, piecewise-linear functional. In this paper algorithms are provided for solving the 'all unit' and the 'incremental' quantity discount, as well as, the 'fixed charge' problems. These algorithms are based upon a branch and bound solution procedure. The branches lead to ordinary transportation problems whose results are obtained utilizing the "cost operator" for one branch and "rim operator" for another branch so that these new problems are not resolved thus reducing computational time. Suitable illustrations and extensions are also provided.
1. Introduction

In the real world, it is a common practice to offer discounts for the purchase of large quantities, and/or for shipment of large volumes of a given commodity. [6,2]

In this paper we analyze quantity discount problems, representing the "all unit" [5] and the "incremental quantity" discounted transportation problems. The details of the procedure are explained considering the 'all unit' quantity discount problem in Section 2, whereas the associated algorithm is presented in Section 3. Certain branch selection procedures and heuristics are provided in Section 4 with corresponding illustrations in Section 5. Based on this algorithm, we provide extensions in Section 6 where the "incremental quantity discounted problem" and the "fixed charge transportation problems" are addressed specifically.

The analysis will focus on the "all unit" quantity discount problem where the methodology concerning the general approach for solving the piecewise linear programming is developed.

Let \( \lambda_{ij}^0 = 0, \lambda_{ij}^1, \lambda_{ij}^2, \ldots, \lambda_{ij}^k, \lambda_{ij}^F \leq m \), where \( \lambda_{ij}^{k-1} < \lambda_{ij}^k \) (for \( k = 1,2,\ldots,r \)) be such that if a quantity \( x_{ij} \) is shipped from source \( i \) to sink \( j \) (\( i = 1,2,\ldots,n \)), \( (j = 1,2,\ldots,n) \) and \( \lambda_{ij}^{k-1} \leq x_{ij} < \lambda_{ij}^k \), then the per unit cost of the \( x_{ij} \) units is \( c_{ij}^k \), and the total cost associated with shipping \( x_{ij} \) units is \( c_{ij}^k x_{ij} \) where \( c_{ij}^k > c_{ij}^{k-1} \). This result is illustrated in figure 1.
This problem may be solved with separable convex programming [11], but solving the transportation problem in this manner has the disadvantage of a large number of constraint equations being necessary.

Our approach is similar to the one suggested by Falk and Soland [4] for solving the general non-convex type math program, but, nevertheless, is more specialized and concerns the special type of non-convex math program, namely, the piecewise linear program.

II. The All Unit Quantity Discount Transportation Type Problem

The transportation type problem with all unit quantity discounts may be formulated as follows:

(1) Minimize \( Z = \sum_{j \in J} \sum_{i \in I} c_{ij}^* x_{ij} \)

(2) Subject to:
\[ \sum_{j \in J} x_{ij} = a_i \quad \text{for } i \in I \]
\[ \sum_{i \in I} x_{ij} = b_j \quad \text{for } j \in J \]

(3) \[ 0 \leq x_{ij} \leq \lambda_{ij}^r \quad \text{for } i \in I \text{ and } j \in J \]

(4) \[ c_{ij}^* = \begin{cases} \lambda_{ij}^0 & \text{if } 0 \leq x_{ij} < \lambda_{ij}^1 \\ \lambda_{ij}^{k-1} & \text{if } \lambda_{ij}^k \leq x_{ij} < \lambda_{ij}^k \\ \lambda_{ij}^r & \text{if } \lambda_{ij}^{r-1} \leq x_{ij} < \lambda_{ij}^r \\ \end{cases} \]

where: \( I = \{1, 2, \ldots, i, \ldots, m\} \) set of sources
\( J = \{1, 2, \ldots, j, \ldots, n\} \) set of sinks
\( K = \{1, 2, \ldots, k, \ldots, r\} \) set of cost intervals
In order to facilitate the presentation, expressions (1) - (5) will be referred to as problem \( P^* \).

The algorithm provided for solving \( P^* \) is basically a branch and bound type similar to the subtour elimination algorithm of the travelling salesman problem \([7]\). Here, instead of eliminating infeasible subtours, we eliminate all infeasibilities due to (5) until complete feasibility is restored.

Let us now define the following "initial Transportation Problem" \( P_0 \) which is given by (1) - (4) of \( P^* \) (note that constraint set (5) is not included), and all \( e_{ij}^* \) are replaced with \( c_{ij}^f \). Since \( c_{ij}^f \) are the minimum values for every \((i,j) \in I \times J\), the solution to problem \( P_0 \) is a "better than optimal" solution.

If, in addition, the solution matrix \( X = [x_{ij}] \) satisfies \( \lambda_{ij}^F \leq x_{ij} < \lambda_{ij}^P \) for every \( i,j \), then the solution of problem \( P_0 \) is the optimum solution to problem \( P^* \).

**Definition 1:** A solution \( X = [x_{ij}] \) to the problem (1) - (4) is said to be "interval feasible" if all \( c_{ij}^* \) used in (1) are implied to be feasible due to the fact that \( x_{ij} \) lies in the feasible interval given by (5).

**Definition 2:** A solution \( X = [x_{ij}] \) to the problem (1) - (4) provided \( \lambda_{ij}^{k-1} \leq x_{ij} < \lambda_{ij}^k \) is said to be the "better than optimal solution" if that solution is an optimal solution to the problem where each \( c_{ij}^* = c_{ij}^f \) in (1) where \( k \leq \lambda \leq 1 \). (Note that \( \lambda \) may be different for different \((i,j) \in (I \times J)\).

It is true that the optimal solution to problem (1) - (5) should be both "better than optimal solution" and "interval feasible." Thus our algorithm presented
here will always possess "better than optimality" criterion and proceed to restore "interval feasibility", similar to any dual algorithm which always has a "better than optimal solution" and approaches "primal feasibility."

III. A General Description of the Algorithm

In the first stage $P_0$ is solved. Thus, if the "better than optimal" solution to $P_0$ is "interval feasible" it is also the optimal solution to $P^*$ given by (1) - (5). If the "better than optimal" solution to $P_0$ is not "interval feasible" to $P^*$ the algorithm proceeds as follows: Let $x_{ij}$ be a value for which "interval feasibility" is violated. More specifically, suppose $x_{ij}$ is "better than optimal" in the interval $\lambda_{ij}^{k-1} \leq x_{ij} < x_{ij}^k$ and its associated cost parameter $c_{ij}^*$ is not equal to $c_{ij}^k$. This condition leads to two branches (subproblems) as follows:

(i) In branch 1 the current $c_{ij}^*$ is replaced by the "interval feasible"

$c_{ij}^k$, and an upper bound restriction in the form of $x_{ij} < x_{ij}^k$ is implied. (Note that it is unnecessary to impose an explicit upper bound. If $x_{ij} \geq x_{ij}^k$ at the optimum, then branch 1 is inferior to branch 2.)

(ii) In branch 2 a lower bound restriction of the form $x_{ij} \geq x_{ij}^k$ is imposed and $c_{ij}^*$ remains unchanged.

The two new transportation problems corresponding to (i) and (ii) are solved (see the solution procedure in Algorithm 1) to determine the lower bounds of $Z$ for all interval-feasible better than optimal solutions in their respective subsets. If the optimal solution corresponding to any one subset is "interval feasible"

We have used the Srinivasan and Thompson [8] algorithm for solving $P_0$.  

that basis from the two subsets but does not exclude any other interval feasible basis. The algorithm converges in a finite number of steps since the total number of bases for the constraint set given by (2) - (5) is finite and each iteration excludes at least one basis. Secondly the branching procedure results in a partition of the interval feasible basic feasible solutions in that subset, and thus the algorithm is expected to be efficient. Third and most importantly each subproblem is not completely resolved. Instead, we apply the "Operator Theory" [8] for the Transportation Problem which is utilized to generate the new solution for each subproblem with minor computational efforts.

Let \( x^*_j \) be one such \( x_{ij} \) where the interval feasibility is violated. (In Section IV we provide a heuristic for the choice of such \( x^*_j \).) First let us consider (i) where \( c^*_j \) is replaced by \( c^k_j \). Let \( c^k_j - c^*_j - \delta > 0 \) (due to (6)) be the positive value to which the current cost \( c^*_j \) is to be increased. The optimal solution to this problem (where the entire data of the problem is unchanged except for the new cost \( c^*_j + \delta \)) is obtained by applying the "cell cost operator" [8,9] \( c^*_j \) to the current problem \( P^* \).

This operation provides the new best optimal solution \( X \) and the optimal total cost \( Z \) for the revised problem.
Now let us consider the second subproblem (ii). Here the only change is the new lower bound imposed on \( x_{st} \), i.e., \( \lambda_{st}^k \leq x_{st} \). Let \( x_{st}' = x_{st} - \lambda_{st}^k \) and \( x_{ij}' = x_{ij} \) for all \( (i,j) \in [I \times J] - \{(s,t)\} \) such that

\[
0 \leq x_{ij}' \leq \infty \quad \text{for all } (i,j) \in [I \times J]
\]

Substituting (7) in the current problem we have

(8) Minimize \[
\sum_{j \in J} \sum_{i \in I} c_{ij}^* x_{ij}' + c_{st}^* \lambda_{st}^k
\]

(9) Subject to \[
\sum_{j \in J} x_{ij}' = a_i \quad \text{for } i \in I \text{ and } i \neq s
\]

(9a) \[
\sum_{i \in I} x_{sj}' = a_s - \lambda_{st}^k
\]

(10) \[
\sum_{i \in I} x_{it}' = b_j \quad \text{for } j \in J \text{ and } j \neq t
\]

(10a) \[
\sum_{i \in I} x_{it}' = b_t - \lambda_{st}^k
\]

(11) \[
0 \leq x_{ij}' \leq \infty
\]

The solution to this new problem (8) - (11) can be obtained by utilizing the "Rim Operator Theory" [8] where a cell rim operator \( \delta \lambda_{st}^k \) is applied by equating \( s = \lambda_{st}^k \) for one row \( s \) and a column \( t \). Note, that in both subproblems we use the "operator theory of parametric programming for the transportation problem" [8,9] for computing the solutions to the branch problems efficiently.
IV. Branch Selection Procedure

In the process of searching for the best optimal solution to problem \( P^* \) we suggest two alternative selection rules: In the first alternative we select that branch which yields the lowest upper bound on the value of the objective function \( z^* \). This strategy is similar to the penalty proposed by Driebeck [3] and Tomlin [10] for solving integer programs. The above procedure tends to postpone the search in less promising branches by concentrating on branches which seem to yield a better solution value. Once our optimal "interval feasible" solution is found, branches yielding inferior upper bounds are eliminated and the number of steps required for solving \( P^* \) are reduced. The process of establishing upper bounds on the current subsets consists of two major stages. In the first stage a heuristic rule is used for selecting the variable to be branched upon in the next step. This stage is followed by a short search which constitutes the second stage. More specifically, let \( \Omega \) represent the current set of cells \( (i, j) \in [I \times J] \) where "interval feasibility" is violated. By introducing a cell \( (i, j) \) as the decision cell for branching, define \( Q_{ij} \) as the infeasibility index such that:

\[
Q_{ij} = (c_{ij}^* - c_{ij})x_{ij}
\]

and determine the most "interval infeasible" index:

\[
Q_{\text{max}} = \max_{(i, j) \in \Omega} Q_{ij} = (c_{st}^* - c_{st})x_{st}
\]

The variable \( x_{st} \) associated with \( Q_{\text{max}} \) is the variable used for branching in the next step. In branch (i) \( c_{st}^* \) is replaced with \( c_{st}^k \), \( x_{st} \) may either stay in the basis or may become nonbasic. If \( x_{st} \) stays in the basis and the current solution is still optimal, the penalty \( p_{st} \) is equal to \( Q_{\text{max}} \) in (12), namely:

\[
p_{st} = (c_{st}^k - c_{st}^*)x_{st}
\]
The heuristic given above takes negligible computational time. However, as a second alternative, if one is prepared for more computations, the following procedure provides a better heuristic. If \( x_{st} \) becomes non-basic, then a non-basic cell \((s,k)\) in the \( s^{th}\) row or a non-basic cell \((k,t)\) in the \( t^{th}\) column replaces \((s,t)\). Let \( T \) be the set of all such non-basic cells in the \( s^{th}\) row and \( t^{th}\) column and let \( a_s \) and \( b_t \) be the dual variables corresponding to row \( s \) and column \( t \), respectively. Let \((u,v) \in T \). Then we could compute a penalty for each \((u,v) \in T \) given by \((c^*_{uv} - a_u - b_v) x_{st} \) and choose the minimum of all such penalties corresponding to every \((u,v) \in T \). Comparing this with the earlier one given in (14) the cell to branch form can be ascertained. It is to be recognized that this procedure involves additional computations and one needs computational experiments for the choice of the best heuristic. Currently, we use the heuristic based on equation (14) and leave the rest for future testing.

Algorithm A1 given below summarizes the above discussion for this quantity discounted non-convex transportation problem.

**Algorithm A1.** Algorithm for finding the optimal solution to the "interval feasible" all unit quantity discounted non-convex transportation problem (1) - (6).

**Initialization:**

**Step 1.** Set up the problem \( P \) as presented in (1) - (4) with \( c^*_{ij} \) in (1) replaced by \( c^r_{ij} \) the smallest cost as given in the \( r^{th}\) interval \( \lambda^r_{ij-1} \leq x_{ij} < \lambda^r_{ij} \leq \). Let \( P_1 \) denote this problem above, and \( N_1 = \emptyset \) denote the set of cells \((i,j)\) that are required to be included in the "final optimal interval feasible" solution. Let \( \Psi_1 = \emptyset \) be the set of all cells \((i,j) \in [I \times J]\) that should be excluded from the "current optimal solution," where the interval in which the basic \( x_{ij} \) lies is the one corresponding to the
current lowest cost $c^*_ij$. Let $X^*_j = \{ x^*_ij \}$ be the optimum solution to $P_j$ with basis $B_j$ and the current optimal cost $z^*_j$. (In this step we solve the transportation problem (1)-(4) and obtain the solution with $c^*_ij = c^*_{ij}$ for $V(i,j)$. Let $S = \{ j \}$ denote the set of problems under consideration and let $m = 1$ denote the total number of problems generated thus far.

**Step 2.** Choose problem $P_k$ for which $z^*_k$ is either the current value of the objective function or the current upper bound on that value depending on the branch selection rule used; it is the smallest for $k \in S$. Let this problem have $c^*_ij = c^u_{ij}$. If $B_k$ is interval feasible, i.e. satisfies the constraint set (5) for every $(i,j) \in B_k$, go to (8). Otherwise go to (3).

**Step 3.** Find the set of cells $A$ where the cells $(i,j)$ in basis $B_k$ violate constraint set (5). Since $P_k$ has costs $c^*_ij = c^u_{ij}$ (the lowest cost) the $x_{ij}$ for $(i,j) \in B_k$ will satisfy (5) if $\lambda^u_{ij} - \lambda^*_ij < \lambda^u_{ij}$ where $u$ is the one determined by $c^*_ij$ for each $(i,j)$. (Note $u$ may be different for different $(i,j)'s$). For each $(i,j) \in A$ find the infeasibility index $Q_{ij}$ given by (12) and select the variable to be branched from as in (13).

Let the cell corresponding to this variable be $(s,t)$.

**Step 4.** Define $P_{mt+1}$ as the problem obtained from $P_k$ by increasing the cost of $c^*_ij$ to $c^u_{st}$, i.e. $P_{mt+1} = P_k \cup \{(s,t)\}$ and let $V_{mt+1} = T_k$. The problem $P_{mt+1}$ and its solution can be obtained from problem $P_k$ by applying "cell cost operator" $\delta c^+_st$ (see Srinivasan and Thompson [8]) to $P_k$ where $\delta = (c^u_{st} - c^*_st) > 0$. (Here $c^u_{st}$ is the new cost due to the fact that $\lambda^u_{st} - \lambda^*_st < \lambda^u_{st}$ and $c^*_st$ is that value which was used as cost for
cell \((s,t)\) in problem \(P_k\). Find the new basis \(B_{m+1}\) and \(Z_{m+1}\) also as per \(\delta_{st}^+\) cost operator \([8]\).

**Step 5.** Define \(P_{m+2}\) as the problem obtained from \(P_k\) by imposing a lower bounded constraint \(x_{st} \geq \lambda_{st}^u\) to the current optimal basis \(B_k\). Set \(Y_{m+2} = Y_k \cup \{(s,t)\}\) and \(\cap_{m+2} = \cap_k\). This solution is obtained by solving the same problem \(P_k\) except that the rim conditions (row and column totals) for \(s\)th row and \(t\)th column will be decreased by a value of \(\lambda_{st}^u\). The solution for \(P_{m+2}\) is obtained by applying the cell rim operator \(\delta_{st}^\pm [8]\). If the new basis \(B_{m+2}\) still contains cell \((s,t)\), change \(x_{st}\) value to \(x_{st}' = x_{st} - \lambda_{st}^u\). The new optimal solution \(Z_{m+2}\) is the optimal solution to problem \(P_{m+2}\) with cost \(\lambda_{st}'\) where \(\lambda_{st}'\) is the same current cost used in problem \(P_k\) for the cell \((s,t)\).

**Step 6.** Denote the basic optimal solutions to \(P_{m+1}\) and \(P_{m+2}\) obtained from \(P_k\) as \(X_{m+1}\) and \(X_{m+2}\) with bases \(B_{m+1}\) and \(B_{m+2}\) respectively. Let \(Z_{m+1}\) and \(Z_{m+2}\) be the corresponding costs for problems \(P_{m+1}\) and \(P_{m+2}\).

**Step 7.** Drop \(k\) from the set \(S\) and include \((m+1)\) and \((m+2)\) in \(S\). Redefine \(m\) as \((m+2)\) and go to (i).

**Step 8.** The optimal solution to the quantity discounted (interval feasible) final problem \((1) - (6)\) is given by \(X_k\) and \(Z_k\) with associated cost \(Z_k\).

Stop.

From a computational viewpoint it is unnecessary to store all the problems \(P_k\) for every \(k \in S\). It is enough if the sets \(\Gamma_k\) and \(Y_k\) are stored for \(k \in S\). With
the original problem $P$ and the cells in $\Omega_k$ and $\Psi_k$ we know the current $P_k$. It is to be noted that in Step 5, while decreasing the row and column totals by $\lambda^{st}$, either one of the resultant row or column totals may become negative. This leads to an infeasible subproblem. In such an eventuality drop that branch from the list of active branches. Also in Step 7 drop $k$ from the set $S$ and include only $(m+1)$ to the set $S$. Next, redefine $m$ as $(m+1)$ and go to step 2.

V. Illustration

In this section we will provide a simple illustration of the above algorithm.

Consider the following three sources ($m = 3$), four destination ($n = 4$) transportation problem, which has quantity discounted transportation costs. The following Table 1 provides the data of the problem for different $c_{ij}^*$ for the three levels of quantity discounts. We provide an example where cell upper bounds are also imposed.

Step 1. Table 2 provides optimal solution for the initial problem. In each cell $(i,j)$ the value of $x_{ij}$ is written in the northeast corner. If a cell is the basis then the corresponding $c_{ij}$ is circled. Those cells which have $x_{ij}$ as their upper bound have their corresponding $c_{ij}$ underlined. Those at zero levels are left out without any entry posted in the northeast corner. The optimal dual variables $u_i$'s for rows and $v_j$'s for columns are given in southeast corner ($N$ denotes large positive $\theta$). It is easy to check that the
<table>
<thead>
<tr>
<th></th>
<th>50</th>
<th>35</th>
<th>60</th>
<th>70</th>
<th>Market Demand</th>
</tr>
</thead>
<tbody>
<tr>
<td>55</td>
<td>[20 &gt; \xi_{55} \leq 0]</td>
<td>[01 &gt; \xi_{55} \leq 0]</td>
<td>[00 &gt; \xi_{55} \leq 0]</td>
<td>[00 \leq \xi_{55} \leq 0]</td>
<td>[00 \leq \xi_{55} \leq 0]</td>
</tr>
<tr>
<td>60</td>
<td>[01 &gt; \xi_{60} \leq 0]</td>
<td>[01 &gt; \xi_{60} \leq 0]</td>
<td>[00 &gt; \xi_{60} \leq 0]</td>
<td>[00 \leq \xi_{60} \leq 0]</td>
<td>[00 \leq \xi_{60} \leq 0]</td>
</tr>
<tr>
<td>60</td>
<td>[01 &gt; \xi_{60} \leq 0]</td>
<td>[01 &gt; \xi_{60} \leq 0]</td>
<td>[00 &gt; \xi_{60} \leq 0]</td>
<td>[00 \leq \xi_{60} \leq 0]</td>
<td>[00 \leq \xi_{60} \leq 0]</td>
</tr>
<tr>
<td>70</td>
<td>[00 \leq \xi_{70} \leq 0]</td>
<td>[00 \leq \xi_{70} \leq 0]</td>
<td>[00 \leq \xi_{70} \leq 0]</td>
<td>[00 \leq \xi_{70} \leq 0]</td>
<td>[00 \leq \xi_{70} \leq 0]</td>
</tr>
</tbody>
</table>

**Table 1**

1. **Quantity Remaining:**
   - 4
   - 3
   - 2
   - 1

2. **Price:**
   - 0
   - \(1_{10} \leq 0\)

3. **Output:**
   - 0
   - \(1_{11} \leq 0\)

4. **Demand:**
   - 0
   - \(1_{12} \leq 0\)
<table>
<thead>
<tr>
<th>Market Demands</th>
<th>70</th>
<th>60</th>
<th>35</th>
<th>60</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demand Capacity</td>
<td>25</td>
<td>60</td>
<td>35</td>
<td>60</td>
</tr>
<tr>
<td>Warehouse Capacity</td>
<td>25</td>
<td>30</td>
<td>40</td>
<td>80</td>
</tr>
<tr>
<td>X_1 is given above; ( z_1 = 945; ) ( S = {1} ) ( m = 1 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
solution is optimal for the costs given. Notice that cells (1,3), (2,2) and (3,1) are not interval feasible.

Step 2. Choose \( p_1; z_1 = 945; B_1 \) is not interval feasible. Hence go to Step 3.

Step 3. The set of interval infeasible cells
\[ A = \{(1,3), (1,2), (3,1)\} . \]
Infeasibility indices are
\[ q_{1,3} = (6-3) * 25 = 25 \]
\[ q_{1,2} = (6-5) * 60 = 60 \]
\[ q_{3,1} = (2-1) * 25 = 25 \]
Thus the variable to branch from is \( x_{22} \).

Step 4. \( x_2 \) is the problem where cell (2,2) is contained in the basis and the cost is changed so that it becomes interval feasible.
\[ \Omega_2 = \Omega \cup \{(2,2)\}; x_2 = \emptyset . \] Problem \( x_2 \)'s solution is obtained by applying cell cost operator \([8]\), \[ \delta C_{22} \] where \( \delta = (6-5) = 1 . \]

Since the current problem is basis preserving, following \([8]\), the new solution becomes \( x^+ = X \) so that \( x_{ij} \) values are not altered, and
\[ z^+ = z + \delta x_{22} = 945 + 1 * 60 = 1005 \]

Form the set \( \Omega = B \setminus \{(2,2)\} = \{(1,1), (1,3), (2,3), (3,1), (3,4)\} \].

Following the notation of \([8]\)
\[ I_p = I_2 = \{1,2,3\}; I_q = \emptyset \]
\[ J_p = \{1,3,4\}; J_q = \{2\} . \]
The maximum extent \( \omega^+ \) to which \( c_{22} \) can be increased without changing the basis structure ("basis preserving") will be as in equation (35) of \([8]\).
\[
\mu^+ = \min \left\{ (e_{ij} - u_i - v_j) \text{ for } (i,j) \in [(I_p \times J_q) \cap LB]\right.
\]
\[
\left. (u_i + v_j - e_{ij}) \text{ for } (i,j) \in [(I_q \times J_p) \cap UB] \right\}.
\]

Now the \((i,j) \in [(I_p \times J_q) \cap LB]\) are cells \((1,2)\) and \((3,2)\)
and \((i,j) \in [(I_q \times J_p) \cap UB] = \emptyset\)

So that \(\mu^+ = \min \{ (6-0-0); (3+2-0) \} = 5 \) occurring at cell \((3,2)\).

Thus the basis remains unchanged. The only change occurs in the optimal duals as given below. (Refer to equation (34) of [8])

\[
u_1^+ = \begin{cases} 
 u_1 + \delta & \text{ for } i \in I_p \\
u_1 & \text{ for } i \in I_q 
\end{cases}
\]

\[
\nu_j^+ = \begin{cases} 
v_j - \delta & \text{ for } j \in J_p \\
v_j & \text{ for } j \in J_q 
\end{cases}
\]

Thus \(u_1^+ = 1; u_2^+ = 6; u_3^+ = -1\)
and \(v_1^+ = 2; v_2^+ = 0; v_3^+ = 2\) and \(\nu^+_q = 6\).

The new tableau for \(P_2\) is presented in Table 3 with \(z_2 = 1005\).
Step 5. \( P_j \) is a new branch created by imposing the bound constraint

\[ 65 \leq x_{22} \leq 60 \text{ on } P_1. \]

This guarantees that \((2,2)\) will be nonbasic. Now \( x_{22} \) becomes 60 - 65 with \( h_2 = 60 - 65 < 0 \). Since demand must be nonnegative, this subproblem is infeasible. Thus, this branch and its associated subbranches are deleted from the list of active branches. Figure 2 summarizes the operations up to this point.

**Figure 2**

\[ P_1 \]

\[ c_{ij} - c_{ij} \]

\[ \{1,3\}, \{2,2\}, \{3,1\} \]

are interval infeasible

Choose \( x_{22} \) for branching

\[ P_2 \]

Apply \( c_{22} \) Operator \( \{1,3\}, \{3,1\} \)

not feasible

\[ P_3 \]

Apply \(-65 R_{22}\)

Problem infeasible

Dropped from future considerations
Step 6. $P_2$, $P_3$ are given. $P_3$ is discarded due to infeasibility. $X_1$ is given in Table 3. $Z_2 = 1005$.

Step 7. (1) is dropped from $S$ and (2) is included in $S$. $m \rightarrow m + 1 \sim 2$. Go to Step 2.

Step 2. $P_2$ is not interval feasible. Go to Step 3.

Step 3. $A = [(1, 1), (3, 1)]$. $Q_{13} = Q_{15} = 25$. Arbitrarily choose $x_{13}$ to branch from.

Step 4. $P_3$ is obtained from $P_2$ by increasing the cost of $c_{13}$ to the interval feasible cost. Following similar cost operation, $1 \cdot c_{13}^+$ application, we see the operation is basis preserving. The new cost $Z_3 = 1005 + 1 \times 25 = 1030$. The optimal duals change. The optimal primal do not change. The resultant optimal tableau is given in Table 4.

Step 5. $P_4$ is obtained from $P_2$ by imposing the lower bounded constraint $27 \leq x_{13} \leq m$. This creates a new $a_1 + 80 - 27 = 53$ and a new $b_3 = 35 - 27 = 8$. Now we apply cell rim operator $68^-$ with $\delta = 27$. From Thm. 2 of [8] the maximum extent $\mu$ that this operator can be applied to be basis preserving is 25. But since $\delta = 27$ we follow the method provided by [8]. The new tableau is given in Table 5 with the new optimal primal and dual solutions.
<table>
<thead>
<tr>
<th>Market Demands</th>
<th>70</th>
<th>60</th>
<th>35</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Warehouse Capacity</td>
<td>25</td>
<td>60</td>
<td>15</td>
<td>25</td>
</tr>
<tr>
<td>u_j</td>
<td>v_i</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>15</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td>60</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>0</td>
<td>25</td>
<td>3</td>
<td>10</td>
<td>5</td>
</tr>
</tbody>
</table>
Note that cell (1,3) has left the basis so that cell (2,4) is in the basis. The optimal cost is \( Z = 920 + 81 = 1011 \).

Since this problem has the least total costs in all pending branches and, nevertheless, it is interval feasible, it is the optimum solution.

Note that the cell (3,1) which was interval infeasible earlier became automatically feasible when the basis change occurred.

The results of branching and bounding are given in Figure 3.
Figure 3

- \( P_1 \)
  \( C_{11}^* = \frac{C_{41}}{k} \)
  \( \Lambda_1^* = \{(1,3),(2,2),(3,1)\} \)

- \( P_2 \)
  Apply 1 \( C_{22}^{++} \)
  \( \Lambda_2 = \{(1,3),(3,1)\} \)
  \( S = [2]; m = 2 \)

- \( X_{13} \)

- \( P_4 \)
  Apply 1 \( C_{13}^+ \)
  \( \Lambda_4 = \{(3,1)\} \)
  \( S = \{2,4\}; m = 2 \)
  \( Z_4 = 1030 \)

- \( P_5 \)
  Apply 27 \( R_{13}^- \)
  \( \Lambda_5 = \{[0]\} \)
  \( S = \{2,4,5\}; m = 2 \)
  \( Z_5 = 1011 \)

STOP
VI. Extensions

(1) The Fixed Charge Problem \[1\]

In this section we outline an algorithm for solving the following fixed charge (transportation) problem:

\[
\begin{align*}
\text{min } Z &= \sum_{i,j} c_{ij} x_{ij} + \sum_{k,l} f_{kl} y_{kl} \\
\text{s.t. } &\sum_{j} x_{ij} = a_i \quad \text{for } i \in I \\
&\sum_{i} x_{ij} = b_i \quad \text{for } j \in J \\
\end{align*}
\]

where \(k \in K, l \in L, i \in I, K \subseteq I, L \subseteq J, \) and \(x_{ij} \geq 0.\)

The method used for solving the fixed charge problem is identical in principle to the one devised in Section IV for the all unit quantity discount type problem.

Here, again, we start by solving a relaxed problem, the solution of which is interval infeasible, then, branch and bound procedure is applied to retain feasibility.

In the following statements the algorithm is summarized:

**Step 1:** Let \(y_{kl} = 0\) for all \(k \in K \) and \(l \in L\) and solve the relaxed transportation problem in (15) – (18).

**Step 2:** If the solution is interval infeasible, i.e., \(x_{kl} > 0, y_{kl} = 0\), select the one variable among all interval infeasible variables for which \(f_{kl}\) is the largest.

**Step 3:** Branch from the variable selected in step 2 into two new problems; in branch 1 increase \(c_{kl}\) to an arbitrary large number making cell \((k,l)\) an
inadmissible cell and solve the new problem using cost operator. In branch 2 introduce a lower bound \( x_{kl} \geq 1 \) and increase the current value of the objective function by \( E_{k1} \).

**Step 4:** Select the one branch with the smallest objective function value among all active branches.

**Step 5:** If the branch selected in step 4 is interval feasible, then stop. This is the optimum. Otherwise go to step 2.

(11) **The Incremental Quantity Discount Problem**

The nonconvex cost structure of this problem is shown in Figure 2. To accommodate such a framework, \( x^k_{ij} \) must be \( \geq \lambda_k \) before \( x^{k+1}_{ij} \) (the amount that can be shipped from \( i \) to \( j \) at a reduced cost \( c^{k+1}_{ij} \)) can be positive, etc.

![Figure 2](image)

Figure 2

The problem can be formulated as follows:

\[
\min \sum_i \sum_j c^k_{ij} x_{ij} + \sum_i \sum_j c^k_{ij} y_{ij}
\]

\[
s.t. \sum_j x_{ij} = a_i \quad \text{for } i \in I
\]

\[
\sum_i x_{ij} = b_j \quad \text{for } j \in J
\]
\( c_{ij}^k = \begin{cases} 
\lambda_{ij}^0 & \text{if } 0 = \lambda_{ij}^0 < x_{ij} < \lambda_{ij}^1 \\
\lambda_{ij}^2 & \text{if } \lambda_{ij}^1 < x_{ij} < \lambda_{ij}^2 \\
\vdots & \text{if } \lambda_{ij}^{r-1} < x_{ij} < \lambda_{ij}^r \\
\vdots & \text{if } \lambda_{ij}^{r} < x_{ij} < \lambda_{ij}^m \\
\end{cases} \)

\( y_{ij}^k = \begin{cases} 
1 & \text{if } \lambda_{ij}^{k-1} < x_{ij} < \lambda_{ij}^k \\
0 & \text{otherwise} \\
\end{cases} \)

and \( x_{ij} \geq 0 \) for all \( i \in I \) and \( j \in J \)

(24) \( f_{ij}^k = \left[ \sum_{v=1}^{k-1} c_{ij}^v (\lambda_{ij}^v - \lambda_{ij}^{v-1}) \right] - c_{ij}^k \lambda_{ij}^{k-1} \) for all \( k = 1, \ldots, r \).

A close examination of the above formulation reveals that the incremental quantity discount problem can be formulated and solved as a generalized fixed charge problem. In the following statements we outline the algorithm:

**Step 1:** Let \( c_{ij}^k = c_{ij} \) and \( y_{ij} = 0 \) for all \( i \in I \) and \( j \in J \) and solve the transportation problem in (19) - (24).

**Step 2:** If the solution is interval infeasible, i.e., there is at least one variable \( x_{ij} \) such that \( \lambda_{ij}^{k-1} < x_{ij} < \lambda_{ij}^k \) and \( y_{ij}^k = 0 \) select the one variable among all interval infeasible variables for which \( f_{ij}^k \) is the largest.

**Step 3:** Branch from the variable selected in step 2 into two new problems: in branch 1 make \( y_{ij}^k = 1 \) (thereby increasing the current value of the
objective function by $s_{ij}^k$ and introduce a lower bound $x_{ij} \geq l_{ij}^k$.
In branch 2 increase $c_{ij}^k$ to $c_{ij}^{k-1}$ and solve the problem in (19)-(24)
using cost operators.

Step 4: Select the branch with the smallest objective function value
from among all active branches.

Step 5: If the solution of the branch selected in step 4 is interval feasible--
stop--this is the optimum. Otherwise go to step 2.

Discussion

In the above examples we showed how one can employ the branch and bound
procedure for solving the piecewise linear programming problem. The algorithm
outlined above yields an efficient procedure in cases where the linear program
has a special structure (as in our examples) because of the important fact
that the special structure is retained throughout the process. Our algorithm
differs from the one suggested by Falk and Soland [4] whose approach may be
characterized by the use of a convex combination of points to approximate the
value of $c_{ij}^k$ over a given range whereas our approach is characterized by the
use of the marginal cost $c_{ij}^k$ at a given range. Falk and Soland's [4] approach
should, then, be considered as a more general framework for solving nonconvex
programs, while our approach is specialized to the case where the nonconvex
program is, nevertheless, piecewise linear.
References