East-West J. of Mathematics: Vol. 13, No 1(2011) pp. 35-49

TRANSPORTATION PROBLEMS WITH TWO SIDED CONTRAINTS ON SUPPLIES AND DEMANDS

Pham Xuan Hinh* and **Tran Vu Thieu**†

**Faculty of Natural Sciences*
Hanoi Teacher Training College, Vietnam
e-mail: pxhinh@yahoo.com

†*Department of Optimization and Control*
Hanoi Institute of Mathematics, Vietnam
e-mail: trvuthieu@yahoo.com

Abstract

The transportation problem with given fixed supplies and demands is well known in the linear optimization theory (see, for instance [1, 3]). It has various applications in the real world and has been widely studied. The purpose of this paper is to develop a new and rather simple algorithm for solving the problem in the case where supplies and demands of stores and destinations have not been fixed, but changed in a given interval. The algorithm relies on reducing the original problem to an ordinary transportation problem with a few bounded variables. This algorithm is a further extension of the one presented in [2] for the case where only supplies or demands can be changed.

1 Problem formulation

Let us first consider the mathematical formulation of the following problem, which will be referred to as transportation problems with two sided-constraints on supplies and demands. We are given a set of m supply points (or stores) from which a good (ex. rice, cement ...) must be shipped. Supply point i $(i = 1, 2, \cdots, m)$ can supply some units of the good, which is in given interval

Key words: Transportation problem, two sided constraints, linear programming with upper bounded variables.
2000 AMS Mathematics Subject Classification: Primary 90C08; Secondary 65K05, 49M29

$[\underline{a}_i, \overline{a}_i]$ $(0 \leq \underline{a}_i \leq \overline{a}_i)$ We wish send the good from the supplies to a set of n demand points (or shops). Demand point j $(j = 1, 2, ..., n)$ need obtain some units of the good, which is in the given interval $\left[\underline{b}_j, \overline{b}_j\right]$ $(0 \leq \underline{b}_j \leq \overline{b}_j)$. Suppose we are known a cost matrix $C = \|c_{ij}\|_{m\times n}$ which determines the cost c_{ij} of shipping a unit of the good from supply point i to demand point j.

If variable x_{ij} is an unknown quantity of the good shipped from supply point i to demand point j and let $\mathbf{x} = \|x_{ij}\|_{m\times n}$, the problem under consideration may be formulated as follows: Minimize objective (or cost) function $f(x)$ defined as

$$f(x) = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \tag{1}$$

subject to

$$\underline{a}_i \leq \sum_{j=1}^{n} x_{ij} \leq \overline{a}_i, i = 1, \cdots, m \tag{2}$$

$$\underline{b}_j \leq \sum_{j=1}^{m} x_{ij} \leq \overline{b}_j, j = 1, \cdots, n \tag{3}$$

$$x_{ij} \geq 0,\ i = 1, \cdots, m, j = 1, \cdots, n \tag{4}$$

where the sum $\sum\limits_{j=1}^{n} x_{ij}$ in (2) represents the total amount of the good shipped from supply point i and the sum $\sum\limits_{i=1}^{m} x_{ij}$ in (3) represents the total amount of the good obtained in demand point j. Let S be the set of all $\mathbf{x} = \|x_{ij}\|_{m\times n}$ satisfying (2) - (4).

We study conditions for solvability of problem (1) - (4). Define

$$\underline{a} = \sum_{i=1}^{m} \underline{a}_i, \overline{a} = \sum_{i=1}^{m} \overline{a}_i, \underline{b} = \sum_{j=1}^{n} \underline{b}_j, \overline{b} = \sum_{j=1}^{n} \overline{b}_j$$

Proposition 1. *In order for $S \neq \emptyset$ it is necessary and sufficient that the following condition holds*

$$[\underline{a}, \overline{a}] \cap \left[\underline{b}, \overline{b}\right] \neq \emptyset \tag{5}$$

Proof. The necessity is obvious because of the sum of all variables x_{ij} in (2) is equal to the sum of all x_{ij} in (3). Conversely, if (5) holds then there exist numbers $a_1, \cdots, a_m$ and $b_1, \cdots, b_n$ such that $\underline{a}_i \leq a_i \leq \overline{a}_i; \underline{b}_j \leq b_j \leq \overline{b}_j$ and

$$a_1 + \cdots + a_m = b_1 + \cdots + b_n \text{ (balance of supply and demand).}$$

Hence $S \neq \emptyset$. In addition, S is bounded because of (2) - (4). Consequently, (5) ensures that problem (1) - (4) has an optimal solution.

2 Reduction to an Equivalent Transportation Problem

To be able to apply the well-known algorithms for solving the transportation problem of linear programming, let us now transform problem (1) - (4) into an equivalent transportation problem that has fixed given supplies and demands and the balance of supply and demand. To do this, we introduce additional variables x_{i0} $(i = 1, 2, \cdots, m)$ and x_{0j} $(j = 1, 2, \cdots, n)$, and consider the following problem, denoted by (P) for short:

$$(P) \qquad f(x) \equiv \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \to \min \tag{6}$$

$$\sum_{j=1}^{n} x_{ij} + x_{i0} = \overline{a}_i, i = 1, 2, \cdots, m \tag{7}$$

$$\sum_{i=1}^{m} x_{ij} + x_{0j} = \overline{b}_j, j = 1, 2, \cdots, n \tag{8}$$

$$x_{ij} \geq 0, i = 1, \cdots, m,\ j = 1, \cdots, n, \tag{9}$$

$$0 \leq x_{i0} \leq e_i = \overline{a}_i - \underline{a}_i, i = 1, \cdots, m \tag{10}$$

$$0 \leq x_{0j} \leq f_j = \overline{b}_j - \underline{b}_j, j = 1, \cdots, n \tag{11}$$

where variable x_{i0} $(i = 1, 2, \cdots, m)$ indicates some amount of the good that remains from the maximum capacity of supply at supply point i (after shipped the good to all the demand points) and x_{0j} $(j = 1, \cdots, n)$ is an amount of the good that is deficient as compared with the maximum need $\overline{b}_j$ of demand point j (after obtained the good from all the supply points)

Proposition 2. *(P) defined by (6) - (11) is equivalent to problem (1) - (4).*

Proof. Let x $= [x_{ij}]_{m\times n}$ be a feasible solution to problem (1) - (4). We define

$$x_{i0} = \overline{a}_i - \sum_{j=1}^{n} x_{ij} \text{ for all } i = 1, 2, \cdots, m$$

and

$$x_{0j} = \overline{b}_j - \sum_{i=1}^{m} x_{ij} \text{ for all } j = 1, 2, \cdots, n$$

that is (7) and (8) hold. It follows from (2) that $0 \leq x_{i0} \leq e_i \equiv \overline{a}_i - \underline{a}_i$ for all $i = 1, \cdots, m$ and $0 \leq x_{0j} \leq f_j \equiv \overline{b}_j - \underline{b}_j$ for all $j = 1, \cdots, n$ So $\{x_{ij}, x_{i0}, x_{0j}\}$ is feasible for (P).

Conversely, if $\{x_{ij}, x_{i0}, x_{0j}\}$ is feasible for (P), it follows from (10) and (11) that $\mathbf{x} = [x_{ij}]_{m\times n}$ satisfies (2) - (4), i.e.$\mathbf{x} = [x_{ij}]_{m\times n}$ is feasible for problem (1) - (4). Therefore, (P) is equivalent to problem (1) - (4).

It is easy to see that (5), which ensures the solvability of problem (1) - (4), is also the solvability condition for (P).

We now show that problem (P) defined by (6) - (11) with assumption that (5) holds is in fact a transportation problem with some upper bounded variables and then describe an appropriate algorithm for this problem.

Summing up constraints (7) for all i we have

$$\sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij} + \sum_{i=1}^{m} x_{i0} = \sum_{i=1}^{m} \bar{a}_i = \bar{a}.$$

Let

$$x_{00} = \sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}, \text{ we get } \sum_{i=0}^{m} x_{i0} = \overline{a} \tag{12}$$

Analogously, summing up constraints (8) for all j we obtain

$$\sum_{j=1}^{n}\sum_{i=1}^{m} x_{ij} + \sum_{j=1}^{n} x_{0j} = \sum_{j=1}^{n} \overline{b}_j = \overline{b}.$$

This implies

$$\sum_{j=0}^{n} x_{0j} = \overline{b} \tag{13}$$

Set $\overline{a}_0 = \overline{b}, \overline{b}_0 = \overline{a}$. By (12) and (13) we can rewrite (P) in the form:

$$(\overline{P}) \qquad f(x) \equiv \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \to \min \tag{14}$$

$$\sum_{j=0}^{n} x_{ij} = \overline{a}_i, i = 0, 1, \cdots, m, \tag{15}$$

$$\sum_{i=0}^{m} x_{ij} = \overline{b}_j, j = 0, 1, \cdots, n, \tag{16}$$

$$x_{ij} \geq 0, i = 0, 1, \cdots, m, j = 0, 1, \cdots, n, \tag{17}$$

$$0 \leq x_{i0} \leq e_i \equiv \overline{a}_i - \underline{a}_i, i = 1, 2, \cdots, m, \tag{18}$$

$$0 \leq x_{0j} \leq f_j \equiv \overline{b}_j - \underline{b}_j, i = 1, 2, \cdots, n, \tag{19}$$

It is worth noting that problem (14) - (19) has the balance of supply and demand:

$$\overline{a}_0 + \overline{a}_1 + \cdots + \overline{a}_m = \overline{b} + \overline{a} \equiv \overline{b}_0 + \overline{b}_1 + \cdots + \overline{b}_n = \overline{a} + \overline{b}.$$

Since (P) has a rather specific structure that is similar to the one of transportation problems unless some bounded variables x_{i0} $(i = 1, 2, ..., m)$ and x_{0j} $(j = 1, 2, ..., n)$ are added, it seems to be unuseful if we simply take use of general techniques of linear programming to treat bounded variables. So in the sequel, exploiting the specific feature of (P) we shall develop a simple and efficient algorithm for solving (P), which actually slightly differs from well-known algorithms for transportation problems. We shall solve $(\bar{P})$ instead of (P).

To solve $(\overline{P})$ we draw a table $\overline{T}$ having $m+1$ rows $i = 0, 1, \cdots, m$ and $n+1$ columns $j = 0, 1, \cdots, n$, where row $i \geq 1$ corresponds to supply point i and column $j \geq 1$ to demand point j; row $i = 0$ writes down the value of variables x_{0j} $(j \geq 1)$, column $j = 0$ writes down the value of variables x_{i0} $(i \geq 1)$. Variable x_{00} is used to put down the total units of the good shipped from all m supply points $i \geq 1$ to all n demand points $j \geq 1$. By (12), (13) and (15) - (17), it can be easily checked that $\overline{T}$ is in fact the table associated with an ordinary transportation problem of size $(m + 1) \times (n + 1)$ of linear programming, which is usually called the transportation table.

We recall that a cell in $\overline{T}$ is an intersection of a row and a column of $\overline{T}$ and denote by (i, j) the cell in row i and column j. For each cell (i, j) in $\overline{T}$ we define a vector, denoted by A_{ij}, of $m + n + 2$ entries with 1 in the i and the $(m + 1 + j)$ entry and zero elsewhere.

We will need the following properties that are known from the transportation table.

Proposition 3. *The system of constraints (15) and (16) of $(\overline{P}\ )$ has the rank $m + n + 1$.*

Proposition 4.. Let $\overline{G}$ be any subset of cells in table $\overline{T}$. Vectors $\{A_{ij} | (i, j) \in \overline{G}\}$ of problem $(\overline{P}\ )$ is linearly independent iff the set of all cells in $\overline{G}$ contains no cycles.

A subset $\overline{G}$ of cells in $\overline{T}$ is said to be chosen if $\overline{G}$ has the two following properties:

(i) $\overline{G}$ has exactly $m + n + 1$ cells that contain no cycles and cell $(0, 0) \in \overline{G}$.

(ii) $(\ \overline{P})$ with additional conditions: $x_{i0} = 0$ or $x_{i0} = e_i$for all $(i, 0) \notin \overline{G}$ $x_{0j} = 0$ or $x_{0j} = f_j$ for all $(0, j) \notin \overline{G}$ and $x_{ij} = 0$ for all $(i, j) \notin \overline{G}, i \geq 1, j \geq 0$ has at least one feasible solution.

Let $\overline{\mathrm{x}} = [x_{ij}]_{(m+1)\times(n+1)}$ be such a solution. Clearly, $0 \leq x_{i0} \leq e_i$ for all $(i,0) \in \overline{G}, 0 \leq x_{0j} \leq f_j$ for all $(0,j) \in \overline{G}, 0 \leq x_{ij}$ for all $(i,j) \in \overline{G}, i \geq 1, j \geq 1$, and vectors A_{ij} for $(i,j) \in \overline{G}$ are linearly independent (by (i) and Proposition 4). Hence, $\overline{\mathrm{x}} = [x_{ij}]_{(m+1)\times(n+1)}$ is a basic feasible solution for $(\overline{P}\ )$. If $\overline{G}$ is a chosen subset of cells in$\overline{T}$ then any cell in $\overline{G}$ is also called chosen.

Given a chosen subset of cells $\overline{G}$ in $\overline{T}$, we can determine for each row $i \geq 1$ a number u_i $(i = 1, 2, \cdots, m)$ and for each column $j \geq 1$ a number $v_j (j = 1, 2, \cdots, n)$ such that

$$\begin{cases} u_i = 0 \quad \text{for all} \quad (i,0) \in \overline{\mathrm{G}} \\ v_j = 0 \quad \text{for all} \quad (0,j) \in \overline{\mathrm{G}} \\ u_i + v_j = c_{ij} \quad \text{for all} \quad (i,j) \in \overline{\mathrm{G}}, i \geq 1, j \geq 1 \end{cases} \tag{20}$$

The numbers u_i and v_j satisfying (20) are called potentials of rows and columns respect to $\overline{G}$, respectively. They can be determined (within a nonzero constant) by solving the system of equations (20), which has a triangle form, as follows. At first, if $(i,0) \in \overline{G}$ let $u_i = 0$ and if $(0,j) \in \overline{G}$ let $v_j = 0$. Then, if $(i,j) \in \overline{G}, j \geq 1$ and u_i was defined but v_j not, set $v_j = c_{ij} - u_i$. At last, if $(i,j) \in \overline{G}, i \geq 1$ and v_j was defined but u_i not, set $u_i = c_{ij} - v_j$. Stop computing whenever the potentials have been found for all rows $i \geq 1$ and all columns $j \geq 1$.

Proposition 5 (Optimality Criteria). *An extreme feasible solution* $\mathbf{x}$ *on a chosen subset* $\overline{G}$ *is optimal for* $(\overline{\mathrm{T}})$ *if the potentials* u_i *and* v_j *(respect to* $\overline{G}$*) satisfy all the following:*

$$\begin{cases} (i)\ u_i \leq 0 \quad \textit{for all} \quad (i,0) \notin \overline{\mathrm{G}}, x_{i0} = 0, \\ (ii)\ u_i \geq 0 \quad \textit{for all} \quad (i,0) \notin \overline{\mathrm{G}}, x_{i0} = e_i, \\ (iii)\ v_j \leq 0 \quad \textit{for all} \quad (0,j) \notin \overline{\mathrm{G}}, x_{0j} = 0, \\ (iv)\ v_j \geq 0 \quad \textit{for all} \quad (0,j) \notin \overline{\mathrm{G}}, x_{0j} = f_j, \\ (v)\ u_i + v_j \leq c_{ij} \quad \textit{for all} \quad (i,j) \notin \overline{\mathrm{G}}, i \geq 1, j \geq 1. \end{cases} \tag{21}$$

3 Algorithm for Solving $[\bar{\mathrm{P}}]$

We are now in a position to describe the algorithm for solving $(\bar{\mathrm{P}})$ defined by (14) - (19).

Step 0 (Find an initial chosen subset of cells and an initial basic feasible solution for $(\bar{\mathrm{P}})$. Assume that $\underline{a} \leq \underline{b} \leq \overline{a}$ (Similarly to case $\underline{b} \leq \underline{a} \leq \overline{b}$). Find the smalleast index k $(1 \leq k \leq m)$ satisfying

$$\overline{a}_1 + \cdots + \overline{a}_k + \underline{a}_{k+1} + \cdots + \underline{a}_m \geq \underline{b} = \underline{b}_1 + \cdots + \underline{b}_n \tag{**}$$

($\underline{b}$ is the sum of minimum demands of all demand points). If $k = m$ then cross out numbers $\underline{a}_{k+1}, \cdots, \underline{a}_m$ on the left of the above inequality.

Define the initial supply amount for each supply point $i = 1, 2, \cdots, m$ as follows:

$$a_i = \begin{cases} \bar{a}_i \quad \text{for all} \quad i = 1. \cdots, k-1, \text{if k} > 1, \\ \underline{b} - (\bar{a}_1 + \cdots + \bar{a}_{k-1} + \underline{a}_{k+1} + \cdots + \underline{a}_m) \quad \text{for} \quad i = k, \\ \underline{a}_i \quad \text{for all} \quad i = k+1. \cdots, m, \quad \text{if k} < \text{m}. \end{cases}$$

Define the initial demand for each demand point $b_j = \underline{b}_j,\ j = 1, 2, \cdots, n$. It is clear that

$$a_1 + a_2 + \cdots + a_m = b_1 + b_2 + \cdots + b_n \equiv \underline{b}$$

Solving a transportation problem of size $m \times n$ with supply vector $a = (a_1, \ldots, a_m)^T$ and demand vector $b = (b_1, b_2, \cdots, b_n)^T$, for example by the potential method [3], obtain an optimal basic solution $\left[x^0_{ij}\right]_{m\times n}$ along with associated potentials u_i, v_j and a chosen subset of cells $G \subset \{(i, j) : 1 \le i \le m, 1 \le j \le n\}$ satisfying

(i) $(i, 0) \notin G$ for all $i, (0, j) \notin G$ for all j, G has $m + n - 1$ cells that contain no cycles,

(ii) $x^0_{ij} = 0$ for all $(i, j) \notin G, i \ge 1, j \ge 1$,

(iii) $u_k = 0, u_i + v_j \le c_{ij}$ for all $(i, j), i \ge 1, j \ge 1$,

(iv) $u_i + v_j = c_{ij}$ for all $(i, j) \in G$.

It can be checked that $\overline{G} = G \cup \{(0, 0), (k, 0)\}$ is a chosen subset of cells in $\overline{T}$ and

$$\overline{\mathrm{x}} = [x_{ij}]_{(m+1)\times(n+1)} =$$

$$= \begin{cases} x_{i0} = 0 \quad \text{for all} \quad i < k, \\ x_{k0} = a_k - \underline{a}_k, \\ x_{i0} = e_i \quad \text{for all} \quad i > k, \\ x_{0j} = \overline{b}_i - \underline{b}_j = f_j \quad \text{for all} \quad j \ge 1, \\ x_{00} = \sum_{i\ge1, j\ge1} x^0_{ij}, \\ x_{ij} = x^0_{ij} \quad \text{for all} \quad (i, j), i \ge 1, j \ge 1, \end{cases}$$

is an basic feasible solution for $(\overline{P})$ and associated potentials of rows and columns are the same u_i and v_j.

Step 1 (Check for optimality). If potentials u_i and v_j satisfy conditions i) - v) in (21) then stop the algorithm and the current basic feasible solution x is optimal. Otherwise, some of the following cases occur:

(i) $u_i > 0$ for some i such that $(i,0) \notin \overline{G}, x_{i0} = 0$,

(ii) $u_i < 0$ for some i such that $(i,0) \notin \overline{G}, x_{i0} = e_i$,

(iii) $v_j > 0$ for some j such that $(0,j) \notin \overline{G}, x_{0j} = 0$,

(iv) $v_j < 0$ for some j such that $(0,j) \notin \overline{G}, x_{0j} = f_j$,

(v) $u_i + v_j > c_{ij}$ for some cell $(i,j) \notin \overline{G}, i \geq 1, j \geq 1$.

Go to Step 2.

Step 2 (Find a new chosen subset of cells). Consider separately one of the five cases:

(i) $u_{i_0} > 0$ for $1 \leq i_0 \leq m, (i_0,0) \notin G, x_{i_0 0} = 0$: Let us denote by C the cycle created by cell $(i_0, 0)$ with some cells in G. Assume that C is of form

$$(i_0,0),(i_1,0),(i_1,j_1),(i_2,j_1),\cdots,(i_k,j_k),(i_0,j_k), k \geq 1, \text{ and } j_1,\cdots,j_k \geq 1 \tag{23}$$

The value of variable $x_{i_0 0}$ will be increased from its present value at 0 and $(i_0, 0)$ becomes the entering (chosen) cell for the next iteration if the new value of $x_{i_0 0}$ doesnt hit its upper bound e_{i_0}. So, we partition the cells of C into two subsets C_+ and C_- by rule: let cell $(i_0, 0)$ in C_+ and cell $(i_1, 0)$ in C_-. The remained cells of C (in columns $j \geq 1$), started from cell (i_1, j_1) are consecutively let in C_+ and in C_-.

(ii) $u_{i_0} < 0$ for $1 \leq i_0 \leq m, (i_0,0) \notin G, x_{i_0 0} = e_{i_0}$: We denote by C the cycle created by cell $(i_0, 0)$ with some cells in G. Assume again that C is of form (23).

The value of variable $x_{i_0 0}$ will be descreased from its present value at the upper bound and $(i_0, 0)$ becomes the entering (chosen) cell for the next iteration if the new value of $x_{i_0 0}$ is still positive. In this case, we partition the cells of C into two subsets C_+ and C_- by another rule: let cell $(i_0, 0)$ in C_- and cell $(i_1, 0)$ in C_+. The remained cells of C (in columns $j \geq 1$), started from cell (i_1, j_1) are consecutively let in C_- and in C_+.

(iii) $v_{j_0} > 0$ for $1 \leq j_0 \leq n, (0,j_0) \notin G, x_{0j_0} = 0$: Define the cycle C created by cell $(0, j_0)$ with some cells in G. Assume that C is of form

$$(0,j_0),(0,j_1),(i_1,j_1),(i_1,j_2),\cdots,(i_k,j_k),(i_k,j_0), k \geq 1, \text{ and } i_1,\cdots,i_k \geq 1 \tag{24}$$

The value of variable x_{0j_0} will be increased from its present value at 0 and $(0, j_0)$ becomes the entering (chosen) cell for the next iteration if the new value of x_{0j_0} doesnt hit its upper bound f_{j_0}. So, we partition the cells of C into two subsets C_+ and C_- by rule: let cell $(0, j_0)$ in C_+ and cell $(0, j_1)$ in C_-. The remained cells of C (in rows $i \geq 1$), started from cell (i_1, j_1) are consecutively let in C_+ and in C_-.

(iv) $v_{j_0} < 0$ for $1 \leq j_0 \leq n, (0, j_0) \notin G, x_{0j_0} = f_{j_0}$: Define the cycle C created by cell $(0, j_0)$ with some cells in G. Assume also that C is of form (24):

The value of variable x_{0j_0} will be descreased from its present value at the upper bound and $(0, j_0)$ becomes the entering (chosen) cell for the next iteration if the new value of x_{0j_0} is still positive. In this case, we partition the cells of C into two subsets C_+ and C_- by a different rule: let cell $(0, j_0)$ in C_- and cell $(0, j_1)$ in C_+. The remained cells of C (in rows $i \geq 1$), started from cell (i_1, j_1) are consecutively let in C_- and in C_+.

(v) $u_r + v_s > c_{rs}$ for $(r, s) \notin G, r \geq 1, s \geq 1$: Define the cycle C created by cell (r, s) with some cellls in G (by successively cross out hanging cells in rows and columns). Consecutively let the cells of C into two subsets C_+ and C_- with the convention that cell $(r, s) \in C_+$. The value of variable x_{rs} will be increased and (r, s) becomes a new chosen cell for the next iteration.

Take one of the indicated cases (Ex. by random rule) and then go to Step 3.

Step 3 (Find the leaving cell and new chosen subset of cells). Define amount of the good h, which is moved on cycle C.

$$h \equiv x_{pq} = \min\{x_{ij} \text{ if } (i,j) \in C_-, e_i - x_{i0} \text{ if } (i,0) \in C_+, f_j - x_{0j} \text{ if } (0,j) \in C_+\} \geq 0 \tag{25}$$

If cell $(p, q) \equiv (i_0, 0)$ and cell $(i_0, 0)$ is taken in case (i) or (ii) of Step 2 or if $(p, q) \equiv (0, j_0)$ and cell $(0, j_0)$ is taken in case (iii) or (iv) of Step 2 then the chosen subset of cells in $\overline{T}$ is unchanged: $\overline{G}' = \overline{G}$. Otherwise, cell (p, q) is excluded from $\overline{G}$ and chosen subset of cells $\overline{G}'$ in $\overline{T}$ which is contains no cycles is defined by

$$G' = (G - \{(p, q)\}) \cup \{(i_0, 0) \text{ or } (0, j_0)) \text{ or } (r, s)\} \tag{26}$$

Step 4 (Update basic solution). The new feasible solution of $(\overline{P})$ is defined by

$$x'_{ij} = \begin{cases} x_{ij}, \text{ if } (i,j) \notin C, \\ x_{ij} + h, \text{ if } (i,j) \in C_+, \\ x_{ij} - h, \text{ if } (i,j) \in C_-. \end{cases} \tag{27}$$

It can be checked that $\overline{x}' = \left[x'_{ij}\right]_{(m+1)\times(n+1)}$ is an basic feasible solution of $(\overline{P})$.

Recompute potentials u_i and v_j (with respect to $(\overline{G}')$) by (20) and go back to Step 1.

An basic feasible solution $\overline{x} = [x_{ij}]_{(m+1)\times(n+1)}$ for $(\overline{P})$ with a chosen subset of cells in $(\overline{G})$ is called nonsingular if $x_{ij} > 0$ for all $(i,j) \in \overline{G}, i \geq 1, j \geq 1$, $0 < x_{i0} < e_i$ for all $(i,0) \in \overline{G}, i \geq 1$ and $0 < x_{0j} < f_j$ for all $(0,j) \in \overline{G}, j \geq 1$ and singular otherwise (that is there exist $x_{ij} = 0$ for $(i,j) \in \overline{G}$ or $x_{i0} = e_i$ for $(i,0) \in \overline{G}$ or $x_{0j} = f_j$ for $(0,j) \in \overline{G}$). In general, it may be able to occur singular feasible solutions while solving $(\overline{P})$ and so is the situation of recycling. Since the recycling is a very rare opportunity in solving $(\overline{P})$, so if there many cells attaining minimum in (25) as (p,q), we can take one of them as an leaving cell. However, we still have the following.

Theorem 1 (Finiteness of the algorithm). *If all extreme feasible solutions of $(\overline{P})$ are nonsingular then the above-described algorithm for solving $(\overline{P})$ terminates after finitely many steps.*

Proof. It is the same as in proof of Theorem 1 in [2], but there are following differences:

When Step 2 is executive, if as the cell dissatisfying optimality criteria (21) we take $(i,0) \notin \overline{G}$ for $u_i > 0, x_{i0} = 0$ (or for $u_i < 0, x_{i0} = e_i$) and if in Step 3 variable x_{i0} changes its value from 0 to e_i (or from e_i to 0) then $\overline{G}' \equiv \overline{G}$. In these cases, the objective functions value is reduced by amount $e_i \times u_i > 0$ (or$-e_i \times u_i > 0$).

Also, if in Step 2 as the cell dissatisfying optimality criteria (21) we take $(0,j) \notin \overline{G}$ for $v_j > 0, x_{0j} = 0$ (or for $v_j < 0, x_{0j} = f_j$) and if in Step 3 variable x_{0j} changes its value from 0 to f_j (or from f_j to 0) then $\overline{G}' \equiv \overline{G}$. In these cases, the objective functions value is reduced by amount $f_j \times v_j > 0$ (or $-f_j \times v_j > 0$).

Consider case $\overline{G}' \neq \overline{G}$. As seen in the proof of Theorem 2 in [2], if in Step 2 the cell dissatisfying optimality criteria (23) is chosen by either of cases (i), (ii), (iii), (iv) or (v) then the objective functions value is reduced by amount $h \times u_i$, $-h \times u_i$, $h \times v_j$, $-h \times v_j$ or $h \times (u_i + v_j - c_{ij}) > 0$, respectively.

Hence, under hypothesis that all basic feasible solutions of $(\overline{P})$ are nonsingular then $h > 0$ by (25) and after each iteration the objective functions value strictly decreases. So, it is not able to occur recycling. Since the total number of all chosen subsets of cells in $\overline{T}$ which are different and no contain cycles is finite, process of solving $(\overline{P})$ must terminate after finitely many steps and finally we obtain an optimal solution of $(\overline{P})$.

4 Ilusstrative Example

Solve problem (1) - (4) with the input data: $m = 3$ and $n = 4$ minimum supply vector $\underline{A} = (50\ \ 60\ 70)$ with sum of its entries $\underline{a} = 180$ and maximum supply vector $\overline{A} = (100\ 120\ 180)$ with sum of its entries $\overline{a} = 400$; minimum demand vector $\underline{B} = (40\ 50\ \ 60\ 70)$ with sum of its entries $\underline{b} = 220$ and maximum demand vector $\overline{B} = (150\ 80\ 100\ 120)$ with sum of its entries $\overline{b} = 450$ and cost matrix

$$C = \begin{bmatrix} 1 & 9 & 5 & 6 \\ 2 & 9 & 8 & 4 \\ 3 & 4 & 2 & 1 \end{bmatrix}.$$

In this example, solvability condition (5) is fulfilled because

$$[180, 400] \cap [220, 450] = [220, 400] \neq \emptyset$$

Step 0. We see that $\underline{a} = 180 < \underline{b} = 220 < \overline{a} = 400$ and the difference of minimum supply and demand is $t \equiv \underline{b} - \underline{a} = 220 - 180 = 40$ So, the smallest index defined by (**) in Step 0 is $k = 1$ and the initial supply amounts for supply points defined by (22) are $a_1 = \underline{a}_1 + t = 50 + 40 = 90 < \overline{a}_1 = 100, a_2 = \underline{a}_2 = 60, a_3 = \underline{a}_3 = 70$.

Solve a transportation of size 3×4 with supply vector $A = (90\ 60\ 70)$ and demand vector $B = (40\ \ 50\ \ 60\ \ 70)$ and cost matrix C, obtaining an optimal solution and associated potentials which is given in Table 1. Steps for solving (P) by the above-described algorithm is summarized in sequential tables.

b_j / a_i	$\underline{40}$	$\underline{50}$	$\underline{60}$	$\underline{70}$	u_i
90	1 • 40	9^-	5 • 50	6	3
$\underline{60}$	2	9	8	4 • 60	3
$\underline{70}$	3	4 • 50	2 • 10	1 • 10	0
v_j	- 2	4	2	1	F =760

Table 1. Chosen subset G and corresponding optimal solution at Step 0

Step 1. G contains cells marked by "•" . Adding to G cell $(0, 0)$ with $x_{00} = 220$, cells $(i, 0), i = 1, 2, 3$ with $x_{i0} = \overline{a}_i - a_i(x_{10} = 10, x_{20} = 60, x_{30} = 110)$ and cells $(0, j), j = 1, 2, 3, 4$ with $x_{0j} = \overline{b}_j - \underline{b}_j = f_j(x_{01} = 110, x_{02} = 30, x_{03} = 40, x_{04} = 50)$, we obtain Table 2.

	j = 0	150	80	100	120	u_i
i = 0	• 220	$\overline{110}$	$\overline{30}$	$\overline{40}$	$\overline{50}$	
100	⊗ 10^+	1 • 40	9	5 • 50^-	6	0
120	$\overline{60}$	2	9	8	4 • 60	0
180	⊙ $\overline{110}^-$	3	4 • 50	2 • 10^+	1 • 10	- 3 < 0
v_j		1	7	5	4	F=760

Table 2. Chosen subset of cells and the 1st basic solution

Cell $(3, 0) \notin G$ with $x_{30} = e_3 = 110, u_3 = -3 < 0$ is the entering (chosen) cell, $h = \min\{x_{30} = 100, x_{13}{=}50, e_1 - x_{10}{=}40\} = 40$ cell (1. 0) is excluded. We obtain Table 3.

	j = 0	150	80	100	120	u_i
i = 0	• 220^+	⊙ $\overline{110}^-$	$\overline{30}$	$\overline{40}$	$\overline{50}$	
100	$\overline{50}$	1 • 40^+	9	5 ⊗ 10^-	6	3
120	$\overline{60}$	2	9	8	4 • 60	3
180	• 70^-	3	4 • 50	2 • 50^+	1 • 10	0
v_j		- 2	4	2	1	F=640

Table 3. Chosen subset of cells and the 2^{nd} basic solution

Cell $(0.1) \notin G$ with $x_{01} = f_1 = 110, v_1 = -2 < 0$ is the entering (chosen) cell, $h = \min\{x_{01} = 100, x_{13} = 10, x_{30} = 70\} = 10, (1.3)$ is the leaving cell. The 3^{rd} basic solution is given in Table 4.

	j = 0	150	80	100	120	u_i
i = 0	• 230^+	• 100^-	$\overline{30}$	$\overline{40}$	$\overline{50}$	
100	$\overline{50}$	1 • 50	9	5 ⊗	6	1
120	$\overline{60}$	2 ⊙	9	8	4 • 60^-	3
180	• 60^-	3	4 • 50	2 • 60	1 • 10^+	0
v_j		0	4	2	1	F=620

Table 4. Chosen subset of cells and the 3^{rd} basic solution

Cell (2.1) with $u_2 + v_1 = 3 > c_{21} = 2$ is the entering cell, $h = \min\{x_{24} = 60, x_{30} = 60, x_{01} = 100\} = 60, (2.4)$ is the leaving cell. The 4^{th} basic solution is given in Table 5.

	j = 0	150	80	100	120	u_i
i = 0	• 290	• 40	$\overline{30}$	$\overline{40}$	$\overline{50}$	
100	$\overline{50}$	1 • 50	9	5	6	1
120	$\overline{60}$	2 • 60	9	8	4	2
180	• 0	3	4 • 50	2 • 60	1 • 70	0
v_j		0	4	2	1	F=560

Table 5. Chosen subset of cells and the 4^{th} basic solution

As the solution in Table 5 has potentials satisfying optimal condition (21) it is an optimal solution of (P). Therefore we found the optimal amount for supply and demand points and the optimal transportation solution as follows (see Table 6).

Supp. \ Dem.	110	$\underline{50}$	$\underline{60}$	$\underline{70}$	u_i
$\underline{50}$	1 • 50	9	5	6	2
$\underline{60}$	2 • 60	9	8	4 • 0	3
$\overline{180}$	3	4 • 50	2 • 60	1 • 70	0
v_j	- 1	4	2	1	F=560

In the listed above tables we used the following symbols "•": chosen cell, "$\odot$": entering cell, "$\otimes$": leaving cell. The bar upper (under) the numbers in "Supp" collumns and "Demand" rows on Tables 1 and 6 show that supply good attains the maximum suply capacity at corresponding supply $\overline{a}_i$ (minmimum $\underline{a}_i$). The bars over the numbers in columns $j = 0$ (rows $i = 0$) in Tables 2 - 5 shows that the value of corresponding variables $x_{i0}(x_{0j})$ hit its upper bound $e_i(f_j)$ ($1^{st}, 2^{nd}$ and 3^{rd} basic solutions).

Remark. Evidently, as compared with algorithms treated linear programming problems with bounded variables, our algorithm is considerable simpler. Calculations are only similar to that of in solving transportation problems of linear programming: only the plus or minus operation and the operation that finds the minimum of nonnegative numbers are used (but not multiplication and division).

5 Conclussion

This paper studies the transportation problem in the case where supplies or / and demands of stores and destinations can be changed in a given interval. The original problem is first transformed into an ordinary transportation problem with a few upper-bounded variables. The algorithm for solving the equivalent problem is presented and an small illustrative example is given.

References

[1] Gass S. I. "Linear Programming", Fifth Edition, McGraw-Hill, New York, 1985.

[2] Hinh P. X. and Thieu T. V. *A new algorithm for solving the ransportation problem with two-sided constrains on supplies or demands,* J. of Applied Mathematics, Vol. V, No 1, 2007, 27 - 42.

[3] Vanderbei R. J. “Linear Programming - Foundation and Extensions”, Third Edition, Springer Verlag, 2008.