

## A Computer Oriented Method for Solving Transportation Problem

Sharmin Afroz and M. Babul Hasan\*

Department of Mathematics, Dhaka University, Dhaka-1000, Bangladesh

(Received : 10 April 2013; Accepted: 29 June 2014)

### Abstract

In this paper, an algorithm and its computer oriented program have been developed for solving transportation programming (TP) reducing it into a linear program (LP). After formulating it into linear programming problems the number of variables becomes large. It then, becomes more difficult and time-consuming if it is done manually with simplex method. By using the computer program the solution can be found in a shorter time. It will be shown that a TP with a large number of variables can be solved in few seconds by using this method. A number of numerical examples are presented to demonstrate the method developed in this research.

**Key Words:** LP, TP, Computer program

### I. Introduction

Linear programming (LP) problems can generally be stated as follows:

$$\begin{aligned} \text{Optimize: } z &= c_1x_1 + c_2x_2 + \dots + c_nx_n \\ \text{Subject to: } a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &= b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n &= b_m \\ x_i &\geq 0, \text{ where } i= 1, 2, \dots, n \\ b_j &\geq 0, \text{ where } j= 1, 2, \dots, m \end{aligned}$$

The characteristics of standard form are given below:

- (i) The objective function is of the optimization i.e. maximization or minimization type.
- (ii) All constraints are expressed as equations.
- (iii) All variables are restricted to be non negative.

The right hand side constant of each constraints is non-negative.

### II. Transportation

In transportation problems (TPs) the objective is to transport various amount of a single homogenous commodity that are initially stored at various origins, to different destinations in such a way that the total transportation cost is minimum. The distinct feature of TPs is that origins and destinations must be expressed in terms of only one kind of unit. These will be shown in this Section.

#### Transportation Table

A specimen of the TP table of m-sources, n-destinations transportation table is given below-

**Table 1. Transportation Table**

		Destination				supply	
		$v_j$	1	2	.....		n
Source	$u_i$	1	$c_{11}$	$c_{12}$	.....	$c_{1n}$	$S_1$
	2	$c_{21}$	$c_{22}$	....	$c_{2n}$	$S_2$	
	.....	...	.....	.....	....	.....	
	m	$c_{m1}$	$c_{m2}$	.....	$c_{mn}$	$S_m$	
Demand			$d_1$	$d_2$	.....	$d_n$	

where,

$u_i$ = multiple of original row i that has been subtracted (directly or indirectly) from original row 0 by the simplex

method during all iterations leading to the current simplex table. Here  $i= 1, 2, \dots, m$   
 $v_j$ = multiple of original row m+j that has been subtracted (directly or indirectly) from original row 0 by the simplex method during all iterations leading to the current simplex table. Here  $j= 1, 2, \dots, n$ .

#### Examples of TPs

In this Section, it will be shown how the TPs minimize the cost using methods of solving TP with the help of numerical examples given as follows.

#### Numerical Example 1

Find the feasible solution of the following transportation problem.

		Warehouse				Supply
		$W_1$	$W_2$	$W_3$	$W_4$	
Factories	$F_1$	14	25	45	5	6
	$F_2$	65	25	35	55	8
	$F_3$	35	3	65	15	16
Demand		4	7	6	13	

Solution: By using North-West corner rule the problem can be solved as follows:

#### Iteration 1

		$V_1=14$	$V_2=25$	$V_3=35$	$V_4=-15$	Supply		
$u_1=0$	4	14	25	-10	45	-20	5	6/2/0
$u_2=0$	-51	65	25	35	-70	55	8/3/0	
$u_3=30$	9	35	52	3	65	13	15	16/1
Demand	4/0	7/5/0	6/3/0	13/0				

#### Iteration 2

		$V_1=14$	$V_2=25$	$V_3=35$	$V_4=37$	Supply		
$u_1=0$	4	14	25	-10	45	32	5	6/2/0
$u_2=0$	-51	65	25	6	35	-18	55	8/6/0
$u_3=-22$	-43	35	3	3	-52	65	13	15
Demand	4/0	7/5/3/0	6/0	13/0				

\* Author for correspondence. e-mail: mbabulhasan@yahoo.com

Iteration 3

	$V_1=14$	$V_2=7$	$V_3=3$	$V_4=5$	Supply
$u_1=0$	14 4	-32 25	-42 45	2 5	6/2/0
$u_2=32$	-19 65	25 2	6 35	-18 55	8/6/0
$u_3=10$	-11 35	3 5	-52 65	11 15	16/11/0
Demand	4/0	7/5/0	6/0	13/11/0	

The basic feasible solution is given below:  
 $x_{11}=4, x_{14}=2, x_{22}=2, x_{23}=6, x_{32}=5, x_{34}=11$   
 Thus the minimum value is  
 $4 \times 14 + 2 \times 5 + 2 \times 25 + 6 \times 35 + 5 \times 3 + 11 \times 15 = 506$

Numerical Example 2

Find the feasible solution of the following transportation problem.

Cost per truck load

Plant	$V_A$	$V_B$	$V_C$	Supply
$U_w$	4	8	8	76
$U_x$	16	24	16	62
$U_y$	8	16	24	77
Demand	72	102	41	

Solution: Similarly, by using North-West corner rule we can solve the problem and find the solution as follows:

The basic feasible solution is given below:

$x_{12}=76, x_{21}=21, x_{23}=41, x_{31}=51, x_{32}=26;$

Thus the minimum cost is

$8 \times 76 + 21 \times 16 + 41 \times 16 + 51 \times 8 + 26 \times 16 = 2424$

III. Formulation of LP Models

In this Section, the formulation of TP into LP will be discussed. By considering Z the total distribution cost and  $x_{ij}$  ( $i=1, 2, \dots, m$ ;

$j=1, 2, \dots, n$ ) the number of units to be distributed from source i to destination j, the LP formulation of TP becomes:

Minimize,  $Z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$

Subject to

$\sum_{j=1}^n x_{ij} \leq s_j, (i = 1, 2, \dots, m)$  (Supply constraints)

$\sum_{i=1}^m x_{ij} \geq d_i, (j = 1, 2, \dots, n)$  (Demand constraints)

$x_{ij} \geq 0, (i=1, 2, \dots, m ; j = 1, 2, \dots, n).$

Applying the process we can formulate the given TP to the LP problems as follows.

Formulation of Numerical Example 1

Minimize,  
 $Z = 14x_{11} + 25x_{12} + 45x_{13} + 5x_{14} + 65x_{21} + 25x_{22}$   
 $+ 35x_{23} + 55x_{24}$   
 $+ 35x_{31} + 3x_{32} + 6x_{33} + 15x_{34}$

subject to,

$x_{11} + x_{12} + x_{13} + x_{14} \leq 6$   
 $x_{21} + x_{22} + x_{23} + x_{24} \leq 8$   
 $x_{31} + x_{32} + x_{33} + x_{34} \leq 16$   
 $x_{11} + x_{21} + x_{31} \geq 4$   
 $x_{12} + x_{22} + x_{32} \geq 7$   
 $x_{13} + x_{23} + x_{33} \geq 6$   
 $x_{14} + x_{24} + x_{34} \geq 13$

and  $x_{ij} \geq 0, (i=1, 2, 3 ; j = 1, 2, 3, 4).$

IV. Algorithm for the Program

In this Section, the combined simplex algorithm for solving LPs is presented.

Step 1: Define the types of constraints. If all are of “ $\leq$ ” types go to step 2. Otherwise go to step 3.

Step 2:

Substep 1: Express in standard form.

Substep 2: State with an initial basic feasible solution in canonical form and set up the initial table.

Substep 3: Use the inner product rule to find the relative profit factors  $c_j$  as follows

$c_j = c_j - z_j = c_j - (\text{inner product of } c_B \text{ and the column corresponding to } x_j \text{ in the canonical system}).$

Substep 4: If all  $c_j \leq 0$ , the current basic feasible solution is optimal and stop calculation. Otherwise select the non-basic variable with most positive  $c_j$  to enter the basis.

Substep 5: Choose the outgoing variable from the basis by minimum ration test.

Substep-6: Perform the pivot operation to get the table and basic feasible solution.

Substep 7: Go to substep 3.

Step 3: At first express the problem in standard form by introducing slack and surplus variables. Then express the problem in canonical form by introducing artificial variables if necessary and form the initial basic feasible solution. Go to substep 3.

Step 4: If any  $c_j$  corresponding to on basic variable is zero, the problem has alternative solution take this column as pivot column and go to substep 5.

Solving the above LP, we see that the optimal basis contains an artificial variable in second iteration. The LP has no feasible solution which is treated by our program efficiency; our program treats this type of LP effectively and shows the basis with artificial variable.

Computer Program

```

vinpt[m_, n_] := Module[{ },
For[i = 1; str = {}, i ≤ m, i++,
  str = Append[str, InputString
    ["Input type of constraints"] ] ];
cstr = {};
t = Table[ Input["Enter row elements"],
  {i, 1, m}, {j, 1, n} ];
tb = Transpose[t]; rhs = Table[ Input
  ["Right hand Constant"], {i, 1, m} ];
ceff = Table[ Input["Cost Vector"],
  {i, 1, n} ]; tbcef = ceff;
For[ i = 1; cstr = {}; bindx = {}, i ≤ m, i++,
  If[ StringMatchQ[ str[[i]], "1" ] = True,
    cstr = Append[cstr, Subscript[S, i] ];
  For[k = 1; s = {}, k ≤ m, k++,
    If[i = k,
      s = Append[s, 1], s = Append[s, 0] ];
    tb = Append[tb, s];
    bindx = Append[bindx, Length[tb] ];
    ceff = Append[ceff, 0];
    tbcef = Append[tbcef, 0],
  ];
  If[ StringMatchQ[ str[[i]], "g" ] = True,
    cstr = Append[cstr, Subscript[S, i] ];
    cstr = Append[cstr, Subscript[A, i] ];
    For[k = 1; s = {}; a = {}, k ≤ m, k++,
      If[ i = k, s = Append[s, -1];
        a = Append[a, 1], s = Append[s, 0];
        a = Append[a, 0] ];
    tb = Append[tb, s]; tb = Append[tb, a];
    bindx = Append[bindx, Length[tb] ];
    ceff = Append[ceff, 0];
    ceff = Append[ceff, -10^10];
    tbcef = Append[tbcef, 0];
    tbcef = Append[tbcef, -M];
    cstr = Append[cstr, Subscript[A, i] ];
    For[k = 1; a = {}, k ≤ m, k++,
      If[ i = k, a = Append[a, 1];
        a = Append[a, 0] ]; tb = Append[tb, a];
    bindx = Append[bindx, Length[tb] ];
    ceff = Append[ceff, -10^10];
    tbcef = Append[tbcef, -M] ]; ]; ];
  For[j = n, j ≥ 1, j--,
    cstr = Prepend[cstr, Subscript[X, j] ];
    tble = Transpose[tb];
    Off[General::spell]
  ]
];

maketble[t_] := Module[{ },
  For[j = 1; coount = {}, j ≤ m + n + pp, j++,
    coount = Append[coount, j] ];
  fb = {"Cj", "Basis", "Cj-Zj"};
  fcj = {"--", "CB", "C̄j"};
  fr = {"RHS", "--", "Z"};
  For[i = 1; cb = {}; tcbf = {};
    cbv = {}; B = {}, i ≤ m, i++,
    For[j = 1, j ≤ m + n + pp, j++,
      If[bindx[[i]] = coount[[j]],
        cb = Append[cb, cstr[[j]] ];
        cbv = Append[cbv, ceff[[j]] ];
        tcbf = Append[tcbf, tbcef[[j]] ];
        B = Append[B, tb[[j]] ], ];
    ]; fb = Insert[fb, cb[[i]], i + 2];
    fcj = Insert[fcj, tcbf[[i]], i + 2];
    fr = Insert[fr, rhs[[i]], i + 2]; ];

```

```

fr = ReplacePart[fr, tcbf.rhs, -1];
B = Transpose[B];
For[i = 1; fbcjr = {};
  cjbar = {}, i ≤ m+n+pp, i++,
  cjbar = Append[ cjbar,
    ceff[[i]] - cbv.Inverse[B].tb[[i]] ];
  fbcjr = Append[ fbcjr,
    (tbcef[[i]] - tcbf.Inverse[B].tb[[i]])
    // Simplify ];];
tbfom = Prepend[ tble, cstr];
tbfom = Prepend[tbfom, tbcef];
tbfom = Append[ tbfom, fbcjr];
tbfom2 = Prepend[Transpose[tbfom], fb];
tbfom2 = Prepend[tbfom2, fcj];
tbfom2 = Append[tbfom2, fr]; hed++; Print["
  Table ", hed, "          "];
Print[];
Print[TableForm [ Transpose[tbfom2],
  TableAlignments → Center,
  TableSpacing -> {1, 3}]];
Print[" -----"];

Print[];
For[i = 1; nofe = 0, i ≤ m, i++,
  If[tcbf[[i]] = -M, nofe = 1]];
If[ Max[cjbar] > 0, Print
  ["Feasible Solution = ", tcbf.rhs],
  Print["Solution Point"];
For[i = 1; k = 0, i ≤ m+n+pp,
  i++, For[j = 1, j ≤ m, j++,
  If[i = bindx[[j]],
  Print[ cb[[j]], " = ", rhs[[j]],
    " (Basic Variable)"]; k = 1 ]];
If[k = 1, , Print[cstr[[i]],
  " = 0 (Non Basic Variable)"]; k = 0];
If[nofe = 0, Print["All  $\bar{C}_j \leq 0$ 
& Optimal Value = ", tcbf.rhs],
  Print["Though all  $\bar{C}_j \leq 0$ ,
but no feasible solution"]];
Off[General::spell]
]

```

```

twobasic[t_] := Module[{},
  If[ Max[cjbar] > 0, p = u[1]; For[j = 1, j ≤ 2, j++,
    For[i = 1; teta = {}, i ≤ m, i++, If[y[[i, p]] > 0,
      teta = Append[teta,  $\frac{rhs[[i]]}{y[[i, p]]}$ ],
      teta = Append[teta,  $10^6$  ]]; ];
  If[Min[teta] ≠  $10^6$ ,
  rr = Position[teta, Min[teta]][[1, 1]];
  rc[j] = Position[teta, Min[teta]],
  Print["Ratio with ", " ", p,
    " th column is not possible"];
  s = 1; pd = u[j]; Goto["end"];
  r[j] = rr; p = u[2];];
  If[r[1] = r[2] && Length[rc[1]] >
    Length[rc[2]], r[1] = rc[1] [[2, 1]];
  If[r[1] = r[2] && Length[rc[1]] <
    Length[rc[2]], r[2] = rc[2] [[2, 1]];

  k = Transpose[{{y[[ r[1], u[1] ]],
    y[[ r[1], u[2] ]]}, {y[[ r[2],
    u[1] ]], y[[ r[2], u[2] ]]}]];
  rh = {rhs[[ r[1] ]], rhs[[ r[2] ]]}];
  For[j = 1, j ≤ 2, j++,
  kop = Transpose[ReplacePart[k, rh, j]];
  rhs[[ r[j] ]] =  $\frac{1}{\text{Det}[k]} * \text{Det}[kop]$ ; ]
  For[i = 1, i ≤ m, i++,
  If[i ≠ r[1] && i ≠ r[2],
  rhs[[i]] =
  rhs[[i]] -
  (y[[i, u[1] ]]* rhs[[ r[1] ]]
  + y[[i, u[2] ]]* rhs[[ r[2] ]]) ] ];

  For[ i = 1, i ≤ m+n+pp, i++,
  yrep = {y[[ r[1], i ]], y[[ r[2], i ]]}];
  For[j = 1, j ≤ 2, j++,
  kop = Transpose[ReplacePart[k, yrep, j]];
  yy[j] =  $\frac{1}{\text{Det}[k]} * \text{Det}[kop]$ ;];
  tble[[ r[1], i ]] = yy[1];
  tble[[ r[2], i ]] = yy[2];
  For[p = 1, p ≤ m, p++, If[p ≠ r[1] && p ≠ r[2],
  tble[[ p, i ]] =
  y[[p, i]] -
  (y[[ p, u[1] ]]
  * yy[1] + y[[ p, u[2] ]]* yy[2]) ] ]
]; Label["end"]; ];

```

```

rowoperation[t_] := Module[{},
  If[Max[cjbar] > 0,
    For[i = 1; teta = {}, i ≤ m,
      i++, If[tble[[i, pcol]] > 0,
        teta = Append[teta,  $\frac{\text{rhs}[[i]]}{\text{tble}[[i, \text{pcol}] ]}$ ],
        teta = Append[teta, 10^6 ]]; ];
  If[Min[teta] = 10^6, Print
    ["Ratio is not possible;
      Unbounded Solution"];
  st = 1; Goto["end"];
  pro = Position[teta, Min[teta]][[1, 1]];
  rhs[[pro]] =  $\frac{\text{rhs}[[pro]]}{\text{tble}[[pro, \text{pcol}] ]}$ ;
  tble[[pro]] = tble[[pro]] *
     $\frac{1}{\text{tble}[[pro, \text{pcol}] ]}$ ;
  For[i = 1, i ≤ m, i++,
    If[i = pro, , rhs[[i]] = rhs[[i]]
      - tble[[i, pcol]] * rhs[[pro]];
    tble[[i]] = tble[[i]] -
      tble[[i, pcol]] * tble[[pro]]; ]
  Label["end"];
  Off[General::spell]
]
onebsop[t_] := Module[{},
  Print["One basic var replacement"];
  Print[]; While[Max[cjbar] > 0,
    pcol = Position[cjbar, Max[cjbar]]
      [[1, 1]]; rowoperation[tble];
    If[st ≠ 1,
      bindx = ReplacePart[bindx,
        coount[[pcol]], pro];
      maketble[tble], Return[] ];
  ]
alter[t_] := Module[{}, For[i = 1;
  nofe = 0, i ≤ m, i++,
  If[bindx[[i]] = 10^10, nofe = 1]];
  nbindx = Complement[coount, bindx];
  For[i = 1; alt = 0, i ≤ Length[nbindx], i++,
  If[cjbar[[nbindx[[i]]]] = 0, alt = 1;
  Print["Alternative Solution"];
  pcol = nbindx[[i]];
  cjbar = ReplacePart[cjbar, 10^6, pcol];
  rowoperation[tble];
  If[st ≠ 1, bindx = ReplacePart[bindx,
    coount[[pcol]], pro];
  maketble[tble], Goto["1st"] ], ];
  Label["1st"]; ];
  If[alt = 0, Print["No Alternative Solution"];
]
main[twobasic_] := Module[{},
  Clear["Context`*"];
  m = Input["No of Rows"];
  n = Input["No of Columns"];
  pp = Input["No of >= constraints"];
  hed = 0; st = 0;
  vinpt[m, n]; maketble[tble];
  d = Input["Choose method \n'1'
    for one basic var \n '2' for two basic var"];
  If[d = 2, twobsop[tble],
    onebsop[tble]; alter[tble]];
  If[s = 1, onebsop[tble]; alter[tble]];
]
Clear[u, r, y]
main[twobasic];

```

### Numerical Examples

In this Section, the TPs presented in Section II will be solved using combined simplex method after converting these problems to LP discussed in Section III.

#### Numerical Example 1

After putting corresponding values for numerical example presented in section III we obtain the results. The final table is shown

**Table 2. Final solution for Numeric Example 1**

--	$C_j$	-14	-25	-45	-5	-65	-25	-35	-55	-35	-3	-65	-15	0	0	0	0	-M	0	-M	0	-M	0	-M	RHS	
$C_B$	Basis	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$	$X_7$	$X_8$	$X_9$	$X_{10}$	$X_{11}$	$X_{12}$	$S_1$	$S_2$	$S_3$	$S_4$	$A_4$	$S_5$	$A_5$	$S_6$	$A_6$	$S_7$	$A_7$	--	
-5	$X_4$	0	1	1	1	-1	0	0	0	-1	0	0	0	0	-1	-1	0	0	-1	1	-1	1	1	-1	1	2
-25	$X_6$	0	0	-1	0	1	1	0	1	0	0	-1	0	0	1	0	0	0	0	0	1	-1	0	0	0	2
-14	$X_1$	1	0	0	0	1	0	0	0	1	0	0	0	1	1	1	0	0	1	-1	1	-1	1	-1	-1	4
-M	$A_4$	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	1	-1	1	-1	1	-1	1	0	0
-3	$X_{10}$	0	1	1	0	-1	0	0	-1	0	1	1	0	0	-1	0	0	0	-1	1	-1	1	0	0	0	5
-35	$X_7$	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	-1	1	0	0	0	6
-15	$X_{12}$	0	-1	-1	0	1	0	0	1	1	0	0	1	0	1	1	0	0	1	-1	1	-1	0	0	0	11
$\bar{C}_j$	$C_j - Z_j$	0	-32	-42	0	-19	0	0	-18	-11	0	-52	0	14-M	46-M	24-M	-M	0	21-M	-21	11-M	-11	9-M	-9	-506	

Solution Point  
 $X_1 = 4$  (Basic Variable)  
 $X_2 = 0$  (Non Basic Variable )  
 $X_3 = 0$  (Non Basic Variable )  
 $X_4 = 2$  (Basic Variable)  
 $X_5 = 0$  (Non Basic Variable )  
 $X_6 = 2$  (Basic Variable)  
 $X_7 = 6$  (Basic Variable)  
 $X_8 = 0$  (Non Basic Variable )  
 $X_9 = 0$  (Non Basic Variable )  
 $X_{10} = 5$  (Basic Variable)  
 $X_{11} = 0$  (Non Basic Variable )  
 $X_{12} = 11$  (Basic Variable)  
 $S_1 = 0$  (Non Basic Variable )  
 $S_2 = 0$  (Non Basic Variable )  
 $S_3 = 0$  (Non Basic Variable )  
 $S_4 = 0$  (Non Basic Variable )

$A_4 = 0$  (Basic Variable)  
 $S_5 = 0$  (Non Basic Variable )  
 $A_5 = 0$  (Non Basic Variable )  
 $S_6 = 0$  (Non Basic Variable )  
 $A_6 = 0$  (Non Basic Variable )  
 $S_7 = 0$  (Non Basic Variable )  
 $A_7 = 0$  (Non Basic Variable )

All  $\bar{C}_j \leq 0$  & Optimal Value = -506  
 No Alternative Solution

Since the program is for maximization problem, the cost vectors are given using negative sign. So the optimal result is negative but it is same with the result which we find using simplex method & methods for solving TP.

*Numerical Example 2*

Similarly, after running the program again we find the results. The final table is shown:

**Table 3. Final solution for Numeric Example 2**

--	$C_j$	-4	-8	-8	-16	-24	-16	-8	-16	-24	0	0	0	0	-M	0	-M	0	-M	RHS	
$C_B$	Basis	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$	$X_7$	$X_8$	$X_9$	$S_1$	$S_2$	$S_3$	$S_4$	$A_4$	$S_5$	$A_5$	$S_6$	$A_6$	--	
-8	$X_2$	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	76
-16	$X_4$	0	0	-1	1	1	0	0	0	-1	0	1	0	0	0	0	0	1	-1	-1	21
-16	$X_6$	-1	0	-1	0	1	0	0	1	0	0	1	1	1	-1	0	0	1	-1	-1	26
-8	$X_7$	1	0	1	0	-1	0	1	0	1	0	-1	0	-1	1	0	0	-1	1	1	51
-M	$A_5$	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	1	-1	1	-1	1	0	0
-16	$X_8$	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	-1	1	1	41
$\bar{C}_j$	$C_j - Z_j$	-4	0	-8	0	0	0	0	0	-16	8-M	24-M	16-M	8-M	-8	-M	0	8-M	-8	-2424	

Solution Point  
 $X_1 = 0$  (Non Basic Variable )  
 $X_2 = 76$  (Basic Variable)  
 $X_3 = 0$  (Non Basic Variable )  
 $X_4 = 21$  (Basic Variable)  
 $X_5 = 0$  (Non Basic Variable )  
 $X_6 = 41$  (Basic Variable)  
 $X_7 = 51$  (Basic Variable)  
 $X_8 = 26$  (Basic Variable)  
 $X_9 = 0$  (Non Basic Variable )  
 $S_1 = 0$  (Non Basic Variable )

$S_2 = 0$  (Non Basic Variable )  
 $S_3 = 0$  (Non Basic Variable )  
 $S_4 = 0$  (Non Basic Variable )  
 $A_4 = 0$  (Non Basic Variable )  
 $S_5 = 0$  (Non Basic Variable )  
 $A_5 = 0$  (Basic Variable)  
 $S_6 = 0$  (Non Basic Variable )  
 $A_6 = 0$  (Non Basic Variable )

All  $\bar{C}_j \leq 0$  & Optimal Value = -2424  
 Alternative Solution

**Table 4. Another final solution for Numeric Example 2**

--	$C_j$	-4	-8	-8	-16	-24	-16	-8	-16	-24	0	0	0	0	-M	0	-M	0	-M	RHS	
$C_B$	Basis	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$	$X_7$	$X_8$	$X_9$	$S_1$	$S_2$	$S_3$	$S_4$	$A_4$	$S_5$	$A_5$	$S_6$	$A_6$	--	
-8	$X_2$	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	76
-24	$X_5$	0	0	-1	1	1	0	0	0	-1	0	1	0	0	0	0	0	1	-1	0	21
-16	$X_6$	-1	0	0	-1	0	0	0	1	1	0	0	1	1	-1	0	0	0	0	0	5
-8	$X_7$	1	0	0	1	0	0	1	0	0	0	0	0	-1	1	0	0	0	0	0	72
-M	$A_5$	0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	1	-1	1	-1	1	0	0
-16	$X_8$	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	-1	1	41
$\bar{C}_j$	$C_j - Z_j$	-4	0	-8	0	0	0	0	0	-16	8-M	24-M	16-M	8-M	-8	-M	0	8-M	-8	-2424	

**Solution Point**

- $X_1 = 0$  (Non Basic Variable )
- $X_2 = 76$  (Basic Variable)
- $X_3 = 0$  (Non Basic Variable )
- $X_4 = 0$  (Non Basic Variable )
- $X_5 = 21$  (Basic Variable)
- $X_6 = 41$  (Basic Variable)
- $X_7 = 72$  (Basic Variable)
- $X_8 = 5$  (Basic Variable)
- $X_9 = 0$  (Non Basic Variable )
- $S_1 = 0$  (Non Basic Variable )
- $S_2 = 0$  (Non Basic Variable )
- $S_3 = 0$  (Non Basic Variable )
- $S_4 = 0$  (Non Basic Variable )
- $A_4 = 0$  (Non Basic Variable )
- $S_5 = 0$  (Non Basic Variable )
- $A_5 = 0$  (Basic Variable)
- $S_6 = 0$  (Non Basic Variable )
- $A_6 = 0$  (Non Basic Variable )

All  $\bar{C}_j \leq 0$  & Optimal Value = -2424

Since, the program is for maximization problem, the cost vectors are given using negative sign. So the optimal result is negative but it is same with the result which we find using other methods.

**V. Discussion and Comparison**

TP are one of the most important and successful applications of quantitative analysis to solve business problems. Generally, the purpose of this problem is to minimize the cost of transporting goods from one location to another which can meet the needs of each arrival. But the methods of solving TP sometimes create difficulties. Lot of calculations were performed to find out the optimum solution. Mistakes may be committed in manual calculations. Most of the methods are time consuming also. After converting these problems into LP and using computer based program which is discussed in this paper, the difficulties have been removed. Using the computer program the desired solution can be found out easily. But the execution time was short. In a short time the solution was found with the help of computer program. It, therefore, be conclude that the computer program is the best process for finding the solution.

**VI. Conclusion**

In this paper, we present an algorithm and its computer oriented program written in the programming language

‘Mathematica’ for solving TP after converting the TP into LP. After formulating the TP into LP the number of variables increases. Then it becomes more difficult and time-consuming if we use manual calculation with simplex method. But using the computer program we can find our solution in a short time. The number of variables is not matter at all in this program. Even for a large number of variables a few seconds is needed for finding solution. In one word, we can say that the solution of TP by converting to LPs and applying the computer technique is one of the best ways.

**References**

1. Asaduzzaman, S.M. and M. Babul Hasan , 2009. A Computer Oriented Combined simplex method for solving linear programming problems, Dhaka University, Journal, **57(2)**, 199-203.
2. Dentzig, G. B. , 1962. Linear Programming and Extension, Princeton University Press, Princeton, N. J, 299-314.
3. Eugere, D., 2001. Schaum’s Outline- Mathematica, McGraw-Hill, New york, 64-90.
4. Frederich S. Hillier, Gerald J. Lieberman, 1995. Introduction of Operations Research, Sixth edition, McGraw-Hill, New York, 350-404.
5. Ganesh Chandra Ray, D., Md. Elias Hossain, 2008. Linear Programming, 3<sup>rd</sup> edition, Titas Publications, Dhaka-1100, 119-292.
6. Gass, S. L., 2004. Linear Programming Method and Application, fifth edition, McGraw-Hill Book Company, New York, 319-350.
7. Gupta, P.K., D.S. Hira, 2005. Problems in Operations Research Principles and Solution, S.Chand& Company LTD., New Delhi-110055, 406-484.
8. Gupta, P.K., Man Mohan, 1997. Linear Programming and Theory of Games, Eighth edition, Sultan Chand & Sons. New Delhi-110002, 343-462.
9. Ravindran, A., Don T. Phillips, James J. Solberg , 1987. Operations Research Principles and Practice, second edition, John Wiley & Sons, New York, 201-250.
10. Reeb, J. and S. Leavengood, 2002. Operations Research (Transportation Problem: A Special case for Linear Programming Problems), Oregon State University, EM8779, \$ 3.50, 610-695.
11. Wolfram, S. , 2000. Mathematica, Addition Wesley Publishing company, Menlo Park, California, New York, 70-80.
12. Winston, W.L., 1993. Operation Research application and Algorithms, third edition, International Thomson Publishing, California, USA, 361-414.