Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 18, Number 2 (2022), pp. 479-499

http://www.ripublication.com/gjpam.htm

MODA – An Innovative Method For Optimality Testing and Optimizing a Solution In Transportation Problems

Dr. R. Murugesan

Associate Professor, Department of Mathematics, St. John's College,
Palayamkottai-627002, India.
(Affiliated to Manonmaniam Sundaranar University)
Tirunelveli-627012, Tamil Nadu, India.

Abstract

In this research article, we propose a very simple and innovative iterative method named MODA (Modified Allocation) for optimality testing and also optimization of a solution obtained by a method in transportation problems. The MODA method has been tested on a number of non-optimal and optimal solutions obtained for many balanced and unbalanced transportation problems. Testing results validate that the proposed MODA method is the ideal one for testing the optimality of an obtained solution and also optimizing that solution, if that's not optimal. Thereby, we obtain the 'tested optimal' solutions to transportation problems. Another added advantage of the proposed method is that it also generates the possible numbers of alternative optimal solutions to a given transportation problem, it they exist to the problem. Further, the proposed MODA method is an alternative to the existing MODI method to test the optimality of a solution and also optimizing it, if it is not optimal.

Keywords: Transportation Problem, Initial Basic Feasible Solution, Optimal Solution, SOFTMIN method, I-SOFT method, MODI method, MODA method.

2010 Mathematics Subject Classification: 90C08

1. INTRODUCTION

The readers of this research article know the fundamentals of Transportation problems (TPs) and the key methods such as NWCM, LCM, VAM, MODI and Stepping Stone [4, 10] available to solve them. In 2012, Abdul Quddoos and et al. [1] introduced a direct method (but actually not direct) called ASM, and its revised version in 2016 [2], to generate an optimal solution directly to a wide range of TPs.

Murugesan R. [8] showed through some illustrative TPs that the ASM method is the method to generate best IBFS only and not a direct method to generate the optimal solution. By identifying some difficulties in the allocation process when tie occurs among certain 0-entry cells, Murugesan R. and Esakkiammal T. [9] improved the existing ASM method and named it as IASM method and showed that the later produces better IBFS than the best IBFS produced by the ASM method.

In 2021, Esakkiammal T. and Murugesan R. [3] proposed an innovative zero allocations approach named SOFTMIN which produces optimal solutions to most of the TPs. In 2022, Murugesan R. [6] established that the SOFTMIN method performs much better than the IASM method, but not a direct method to produce optimal solution to any given TP.

We further analyzed the process of allocation due to the SOFTMIN method on the near optimal solutions obtained for some 'More Challenging' TPs, and identified that very few changes made in the allocation process have improved the solution. This resulted in the" Improved SOFTMIN" (or briefly I-SOFT) method [7]. As for as our knowledge/search is concerned so far, no competing methods for generating best initial basic feasible solution (IBFS) on the identified some 'More Challenging' TPs are not available in the literature and thereby, the I-SOFT method may be the best one to produce the best IBFS to a given TP.

In this research article we have proposed an innovative method named MODA (Modified Allocation) which tests the optimality of a solution and also optimizes the solution, if it is not optimal.

The paper is organized as follows: Section 1 – Briefs the introduction. Section 2 – Presents the algorithm of the proposed MODA method. Section 3 –Illustrates two 'more challenging' TPs by the proposed method. Section 4 – Lists a set of 14 "More Challenging" TPs. Section 5 – Discusses on the results obtained. Section 6 – Draws the conclusion.

Basic cell and Non-basic cell

A cell in a transportation table (TT) is said to be a 'basic cell' if it is having some allocated quantity. The allocated quantity may be positive or zero. The other cells are called 'non-basic cells'. A TT with size m × n will have at most (m+n-1) basic cells and the remaining cells are non-basic.

Non-degenerate BFS and Degenerate BFS

A basic feasible solution (BFS) to a TP is said to be 'non-degenerate' if it contains exactly (m+n-1) numbers of basic cells. A BFS is said to be 'degenerate' if it contains less than (m+n-1) numbers of basic cells.

Loop in a TT

A 'loop' in a transportation table is an ordered set of even numbers (≥ 4) of cells having only one non-basic cell and the remaining basic cells. The cells in a loop are called ‘corner cells’.

2. ALGORITHM FOR THE PROPOSED ‘MODA’ METHOD

The term MODA has been coined from the first three letters of the word ‘Modified’ and the first one letter of the word ‘Allocation’. MODA is an iterative method which can be used for testing the optimality of an initial basic feasible solution (IBFS) and also optimize the IBFS, if it is not optimal, for transportation problems. The innovative way of improving a non-optimal solution to an optimal solution by MODA method is based on redistributing the allocation available at a currently allocated cell (basic cell) with largest ‘unit transportation cost’ (UTC) to another un-allocated cell (non-basic cell) and its subsequent induced reallocations. The algorithm of the proposed MODA method consists of two stages. In Stage #1, an IBFS is obtained to the given TP. In Stage #2, optimality testing of the obtained IBFS and also optimizing it, if it is not optimal, is carried out.

We use the following notations and abbreviations in the development of the algorithm of the MODA method:

m × n	– Size of the unit cost matrix of the given TP
TT	– Transportation table
BTP	– Balanced transportation problem
UTP	– Unbalanced transportation problem
IBFS	– Initial basic feasible solution
UTC	– Unit transportation cost
C_{ij}	– UTC available at the cell (i, j)
TTC	– Total transportation cost
Z	– TTC
Z^*	– Minimum TTC
NCC	– Net Cost Change
I-index	– Improvement index

STAGE #1: OBTAIN AN IBFS TO THE GIVEN TP

For the given TP, first find an IBFS with its total transportation cost Z using any available method in TPs. We use the I-SOFT method [7] to find an IBFS because at present day it has been identified and established as the best method to find the best IBFS to TPs.

STAGE #2: TEST THE OPTIMALITY OF THE OBTAINED IBFS

Step 1: Construct the current solution table

Consider the transportation table (TT) en-squared with the obtained allocations (solution) as the current solution table. Also, compute the corresponding TTC.

Step 2: Ensure the Non-degeneracy condition

Ensure the numbers of basic cells in the TT exactly equal to (m+n-1).

Step 3: Identify the (next) basic cell with the largest UTC

Among the (m+n-1) basic cells, identify the one say (i, j) having the largest UTC C_{ij}.

Step 4: Trace all the possible loops starting and ending at the identified basic cell (i, j)

Trace all possible loops starting and ending at the identified basic cell (i, j) and passing through only one non-basic cell

Step 5: Compute the NCC value for the traced loops

Consider one traced loop. Mark with a + sign and a – sign alternatively at each of the 'corner cells' of the considered loop, starting from the non-basic cell (k, l) in it. Compute the 'effect on cost' of the considered loop by adding together the original UTC found in each corner cell containing a + sign and then subtracting the original UTC found in each corner cell containing a – sign. This effect on cost is called the 'net cost change' (NCC) value for the considered loop. If the identified basic cell (i, j) has more than one loop, then compute the 'NCC value' for each of the possible loops traced. The minimum among the NCC values is called the 'Improvement Index' or simply called 'I-index' due to the identified basic cell (i, j).

Step 6: Test the optimality

The computed I-index for the identified basic cell (i, j) may be negative, zero or positive. If we implement the loop with the negative I-index only will decrease the total transportation cost (TTC) further.

1. If the I-index computed for each of the (m+n-1) basic cells is non-negative, then we have reached the 'optimal level' and hence the current solution is optimal. Write the optimal solution and compute the corresponding minimum TTC (Z^*).
2. If the I-index computed for an identified basic cell is negative, then the current solution is not optimal and the solution can be improved further by implementing the exact loop corresponding to the negative I-index.
3. If the I-index computed for an identified basic cell (i, j) with UTC C_{ij} is non-negative then go to Step 3 to identify the next basic cell (p, q) having the largest UTC C_{pq} such that $C_{pq} \leq C_{ij}$.

ALTERNATIVE OPTIMAL SOLUTION

At the ‘optimal level’, if the I-index of a basic cell is zero, then this indicates that the given TP has an alternative optimal solution. By implementing the loop corresponding to the I-index zero (NCC value 0), we can get the alternative optimal solution to the given TP.

SAMENESS BETWEEN ‘MODI’ AND ‘MODA’

One can verify the following two statements regarding the sameness between the existing MODI (Modified Distribution) method and the proposed MODA (Modified Allocation) method:

1. For a given non-optimal solution of a TP, the MODI method and the MODA method takes the same numbers of iterations to reach the ‘optimal level’.
2. For a given non-optimal solution of a TP, the MODI method and the MODA method identifies the same identical ‘non-basic cell’ to enter as a ‘basic cell’ during the corresponding iteration.

DIFFERENCE BETWEEN ‘MODI’ AND ‘MODA’

1. In a TT with size m × n, there are mn cells in total. Out of these, (m+n-1) numbers are basic and the remaining mn - (m+n-1) are non-basic. Note that, [mn-(m+n-1)] > (m+n-1), when m, n > 3. In MODI method one has to compute the net evaluation (opportunity cost) corresponding to each of the [mn - (m+n-1)] non-basic cells in order to identify the appropriate one to enter as a basic one. But in the proposed MODA method, it is enough to test a very few of the (m+n-1) basic cells (or at most (m+n-1) basic cells) to identify the ‘leaving’ basic cell into a non-basic one and an ‘entering’ non-basic cell into a basic one. Due to this the computational time is considerably less in the proposed MODA method.
2. In MODI method, we trace a loop starting and ending at the identified non-basic cell with the most negative net evaluation. But in MODA method, we trace a loop starting and ending at the identified basic cell having the computed most negative I-index.

Tested optimal solution

An optimal solution is regarded as ‘tested optimal’ if it comes out of an algorithm after testing its optimality.

Default optimal solution

An optimal solution is regarded as ‘default optimal’ if it comes out of an algorithm as optimal directly. Default optimal is optimal by default.

IMPORTANT NOTE

1. In transportation problems, the optimal solutions generated through either the MODI method or Stepping Stone method [4, 10] are ‘tested optimal’. So far, no direct method has been developed to produce the ‘default optimal’ solutions

directly to transportation problems. This is the fact from the article titled Neeya? Naana? due to Murugesan R. [8]

2. For the assignment problems, the optimal solutions generated by the NILA technique [5] are 'tested optimal' and that of generated by the 'Hungarian' method and the 'Mantra' technique [5] are 'default optimal' by nature. The latter two are the 'direct' methods which produce 'default optimal' solutions to assignment problems. It is the design (divine!) specialty of the two methods.

3. NUMERICAL ILLUSTRATION

Suitable illustrative explanation helps the readers to understand the algorithm of the proposed MODA method in a better way. Keeping in mind, two 'more challenging' TPs from the literature are illustrated.

Example-1: Consider the following cost minimization type BTP with five sources and five destinations, as given in Table 1.

Table 1: The given BTP

Sources	Destinations					Supply
	D1	**D2**	**D3**	**D4**	**D5**	
S1	8	8	2	10	2	**40**
S2	11	4	10	9	4	**70**
S3	5	2	2	11	10	**35**
S4	10	6	6	5	2	**90**
S5	8	11	8	6	4	**85**
Demand	**80**	**55**	**60**	**80**	**45**	**320**

SOLUTION BY THE PROPOSED 'MODA' METHOD

Stage #1: Obtain an IBFS

In Stage #1, we solve the given BTP by using the I-SOFT method and obtain the near optimal solution table. This is shown in Table 2 and the corresponding computed total transportation cost (TTC) Z is shown in Table 3.

Stage #2: Optimizing the obtained solution by the proposed MODA method

FIRST ITERATION

Step 1: Construct the current solution table

Consider the transportation table (TT) en-squared with the obtained allocations (solution) as the current solution table. This is shown in Table 2 and the corresponding TTC of Z = $1640, is shown in Table 3.

Table 2: The Near Optimal Solution table obtained by the I-SOFT method

Sources	Destinations					Supply
	D1	D2	D3	D4	D5	
S1	8	8	**40** 2	10	2	**40**
S2	**15** 11	**55** 4	10	9	4	**70**
S3	**15** 5	2	**20** 2	11	10	**35**
S4	10	6	6	**45** 5	**45** 2	**90**
S5	**50** 8	11	8	**35** 6	4	**85**
Demand	**80**	**55**	**60**	**80**	**45**	**320**

Table 3: The computed TTC

Basic cells in order	Allocated Quantity	UTC in the basic cell	Quantity × UTC
(S1, D3)	40	2	080
(S2, D1)	15	**11**	165
(S2, D2)	55	4	220
(S3, D1)	15	5	075
(S3, D3)	20	2	040
(S4, D1)	45	**10**	450
(S4, D5)	45	2	090
(S5, D1)	05	8	040
(S5, D4)	80	6	480
Total transportation cost (Z)			**1640**

Step 2: Ensure the Non-degeneracy condition

The obtained solution is non-degenerate as there are exactly 9, that is, (m+n-1) basic cells.

Step 3: Identify the basic cell with the largest UTC

Among the 9 basic cells, the basic cell (S2, D1) is having the largest UTC of $11.

Step 4 & 5: Trace all the possible loops starting and ending at the basic cell (S2, D1) and compute its I-index

Trace all possible loops starting and ending at the identified basic cell (S2, D1) and passing through only one non-basic cell.

Loop 1 = {(S2, D1), (S2, D3), (S3, D3), (S3, D1), (S2, D1)}, NCC = 10-11+5-2 = +2

Loop 2 = {(S2, D1), (S2, D4), (S5, D4), (S5, D1), (S2, D1)}, NCC = 9-11+8-6 = **0**

Loop 3 = {(S2, D1), (S2, D5), (S4, D5), (S4, D1), (S2, D1)}, NCC = 4-11+10-2 = +1

Loop 4 = {(S2, D1), (S2, D2), (S3, D2), (S3, D1), (S2, D1)}, NCC = 2-5+11-4 = +4

Loop 5 = {(S2, D1), (S2, D2), (S4, D2), (S4, D1), (S2, D1)}, NCC = 6-10+11-4 = +3

Loop 6 = {(S2, D1), (S2, D2), (S5, D2), (S5, D1), (S2, D1)}, NCC = 11-8+11-4 = +10

I-index = Min{+2, 0, +1, +4, +3, +10} = 0

Step 6: Test the optimality

As the I-index for the identified basic cell (S2, D1) is zero, by the implementation of the corresponding Loop 2, there will be neither a decrease nor an increase in Z value. Therefore, we go to Step 3 to consider the next basic cell having the UTC smaller than or equal to $11.

Step 3: Identify the next basic cell with the largest UTC ≤$11

The next basic cell is (S4, D1) having the UTC of $10.

Step 4 & 5: Trace all possible loops starting and ending at the basic cell (S4, D1) and compute its I-index

Trace all possible loops starting and ending at the identified basic cell (S4, D1) and passing through only one non-basic cell.

Loop 1 = {(S4, D1), (S4, D3), (S3, D3), (S3, D1), (S4, D1)}, NCC = 6-10+5-2 = **-1**

Loop 2 = {(S4, D1), (S4, D4), (S5, D4), (S5, D1), (S4, D1)}, NCC = 5-10+8-6 = **-3**

Loop 3 = {(S4, D1), (S4, D2), (S2, D2), (S2, D1), (S4, D1)}, NCC = 6-4+11-10 = +3

Loop 2 = {(S4, D1), (S4, D3), (S3, D3), (35, D1), (S4, D1)}, NCC = 6-2+5-10 = **-1**

I-index = Min{-1, -3, +3, -1} = -3

Step 6: Test the optimality

As the I-index for the identified basic cell (S4, D1) is negative, the current IBFS is not optimal. By implementing the corresponding Loop 2, definitely there will be a decrease in Z value. By implementing the Loop 2 we obtain the modified allocation table, as shown in Table 4, and the corresponding improved TTC of Z = $1545, as shown in Table 5.

Table 4: The Modified Allocation table by the MODA method

Sources	Destinations					Supply
	D1	D2	D3	D4	D5	
S1	8	8	**40** 2	10	2	**40**
S2	11	**55** 4	10	9	**15** 4	**70**
S3	**15** 5	2	**20** 2	11	10	**35**
S4	10	6	6	**60** 5	**30** 2	**90**
S5	**65** 8	11	8	**20** 6	4	**85**
Demand	**80**	**55**	**60**	**80**	**45**	**320**

Table 5: The improved solution with the computed TTC

Basic cell in order	Allocated Quantity	UTC in the basic cell	Quantity × UTC
(S1, D3)	40	2	080
(S2, D1)	15	**11**	165
(S2, D2)	55	4	220
(S3, D1)	15	5	075
(S3, D3)	20	2	040
(S4, D4)	45	5	225
(S4, D5)	45	2	090
(S5, D1)	55	8	440
(S5, D4)	35	6	210
Total transportation cost (Z)			**1545**

SECOND ITERATION

Step 2: Ensure the Non-degeneracy condition

The obtained improved solution is non-degenerate as there are exactly 9, that is, (m+n-1) basic cells.

Step 3: Identify the basic cell with the largest UTC

Among the 9 basic cells, the basic cell (S2, D1) is having the largest UTC of $11.

Step 4 & 5: Trace all possible loops starting and ending at the basic cell (S2, D1) and compute its I-index

Trace all possible loops starting and ending at the identified basic cell (S2, D1) and passing through only one non-basic cell.

Loop 1 = {(S2, D1), (S2, D3), (S3, D3), (S3, D1), (S2, D1)}, NCC = 10-11+5-2 = +2

Loop 2 = {(S2, D1), (S2, D5), (S4, D5), (S4, D4), (S5, D4), (S5, D1), (S2, D1)},

NCC = 4-11+8-6+5-2 = **-2**,

Loop 3 = {(S2, D1), (S2, D2), (S3, D2), (S3, D1), (S2, D1), NCC = 2-5+11-4 = +4

Loop 2 = {(S2, D1), (S2, D2), (S5, D2), (S5, D1), (S2, D1), NCC = 11-8+11-4 = +10

I-index = Min{+2, -2, +4, 10} = -2

Step 6: Test the optimality

As the I-index for the identified basic cell (S2, D1) is negative, the current solution is also not optimal. By implementing the corresponding Loop 2, definitely there will be a decrease in Z value. By implementing the Loop 2 we obtain the modified allocation table, as shown in Table 6, and the corresponding improved TTC of $1475, as shown in Table 7.

Table 6: The Modified Allocation table due to the MODA method

Sources	Destinations				
	D1	D2	D3	D4	**Supply**
S1	3	48	14	**24** 2	**24**
S2	**6** 4	**10** 2	30	**8** 10	**24**
S3	36	**2** 8	12	12	**02**
S4	0	0	**3** 0	**12** 0	**15**
Demand	**6**	**12**	**3**	**44**	**320**

Table 7: The improved Solution (optimal) with the computed TTC

Basic cells in order	Allocated Quantity	UTC in the basic cell	Quantity × UTC
(S1, D3)	40	2	080
(S2, D2)	55	4	220
(S2, D5)	15	4	060
(S3, D1)	15	5	075
(S3, D3)	20	2	040
(S4, D4)	60	5	300
(S4, D5)	30	2	060
(S5, D1)	65	**8**	520
(S5, D4)	20	6	120
		Minimum TTC (Z^*)	**1475**

THIRD ITERATION

Step 2: Ensure the Non-degeneracy condition

The obtained solution is non-degenerate as there are exactly 9, that is, (m+n-1) basic cells.

Step 3: Identify the basic cell with the largest UTC

Among the 9 basic cells, the basic cell (S5, D1) is having the largest UTC of $8.

Step 4 & 5: Trace all possible loops starting and ending at the basic cell (S5, D1) and compute its I-index

Trace all possible loops starting and ending at the identified basic cell (S5, D1) and passing through only one non-basic cell.

Loop 1 = {(S5, D1), (S5, D3), (S3, D3), (S3, D1), (S5, D1)}, NCC = 8-2+5-8 = +3

Loop 2 = {(S5, D1), (S5, D4), (S4, D4), (S4, D1), (S5, D1)}, NCC = 10-8+6-5 = +3

I-index = +3

Step 6: Test the optimality

As the I-index for the identified basic cell (S5, D1) is positive, by the implementation of the corresponding Loop, there will be no decrease in Z value. Therefore, we go to Step 3 to identify the next basic cell having the UTC smaller than or equal to $8.

Step 3: Identify the next basic cell with the largest UTC

The next basic cell is (S5, D4) having the UTC of $6.

Step 4 & 5: Trace all possible loops starting and ending at the basic cell (S5, D4) and compute its I-index

Trace all possible loops starting and ending at the identified basic cell (S5, D4) and passing through only one non-basic

Loop 1 = {(S5, D4), (S5, D5), (S4, D5), (S4, D4), (S5, D4)}, NCC = 4-2+5-6 = +1

Loop 2 = {(S5, D4), (S4, D4), (S4, D1), (S5, D1), (S5, D4)}, NCC = 10-8+6-5 = +3

Loop 3 = {(S5, D4), (S3, D4), (S3, D1), (S5, D1), (S5, D4)}, NCC = 11-5+8-6 = +8

I-index = +1

Hence, improvement is not possible by implementing the corresponding loop.

Also, we can see that I-index of each of the remaining basic cells is positive. Therefore, we stop the improvement process.

Step 6: Test the optimality

Thus, from each of the basic cells, the traced loops will not improve the current solution further. This indicates that the current solution, shown either in Table 6 or in Table 7 with the TTC of $Z^* = \$1475$, is optimal only.

Example-2: Consider the following cost minimization type UTP with three sources and four destinations, as given in Table 8.

Table 8: The given UTP

Sources	Destinations				Supply
	D1	**D2**	**D3**	**D4**	
S1	3	48	14	2	**24**
S2	4	2	30	10	**24**
S3	36	8	12	12	**02**
Demand	**6**	**12**	**3**	**44**	**65 \ 50**

SOLUTION BY THE PROPOSED 'MODA' METHOD

Stage #1: Obtain an IBFS

In Stage #1, we solve the given UTP by using the I-SOFT method and obtain the near optimal solution table as shown in Table 9 and the associated total transportation cost (TTC) Z as shown in Table 10. Note that, the dummy source S4 has been added to satisfy the extra demand of 15 units. The UTC for each of the dummy cells is assumed to be zero.

Stage #2: Optimizing the obtained solution by the proposed MODA method

FIRST ITERATION

Step 1: Construct the current solution table

Consider the transportation table (TT) en-squared with the obtained allocations (solution) as the current solution table. This is shown in Table 8 and the corresponding computed TTC of Z = $188 is shown in Table 10.

Table 9: The Near Optimal Solution table by the I-SOFT method

Sources	Destinations D1	D2	D3	D4	Supply
S1	3	48	14	**24** 2	**24**
S2	**6** 4	**12** 2	30	**6** 10	**24**
S3	36	8	**2** 12	12	**02**
S4	0	0	**1** 0	**14** 0	**15**
Demand	**6**	**12**	**3**	**44**	**320**

Table 10: The computed TTC

Basic cells in order	Allocated Quantity	UTC in the basic cell	Quantity × UTC
(S1, D4)	24	2	48
(S2, D1)	06	4	24
(S2, D2)	10	2	20
(S2, D4)	08	**10**	80
(S3, D2)	02	8	16
(S4, D3)	03	0	00
(S4, D4)	12	0	00
Total transportation cost (Z)			**188**

Step 2: Ensure the Non-degeneracy condition

The obtained solution is non-degenerate as there are exactly 7, that is, (m+n-1) basic cells.

Step 3: Identify the basic cell with the largest UTC

Among the 7 basic cells, the basic cell (S2, D4) is having the largest UTC of $10.

Step 4 & 5: Trace all possible loops starting and ending at the basic cell (S2, D4) and compute its I-index

Trace all possible loops starting and ending at the identified basic cell (S2, D4) and passing through only one non-basic cell.

Loop 1 = {(S2, D4), (S4, D4), (S4, D3), **(S3, D3)**, (S3, D2), (S2, D2), (S2, D4)},

NCC = 12-0+0-10+2-8 = **-4**

Loop 2 = {(S2, D4), **(S3, D4)**, (S3, D2), (S2, D2), (S2, D4)}, NCC = 12-10+2-8 = **-4**

Loop 3 = {(S2, D4), (S4, D4), (S4, D2), (S2, D2), (S2, D4)}, NCC = 0-2+10-0 = +8

Loop 4 = {(S2, D4), (S4, D4), (S4, D1), (S2, D1), (S2, D4)}, NCC = 0-4+10-0 = +6

Loop 5 = {(S2, D4), (S1, D4), (S1, D2), (S2, D2), (S2, D4)}, NCC = 48-2+10-2 = +54

Loop 6 = {(S2, D4), (S1, D4), (S1, D1), (S2, D1), (S2, D4)}, NCC = 3-4+10-2 = +7

I-index = Min{-4, -4, +8, +6, 54, +7} = -4

Step 6: Test the optimality

As the I-index due to the identified basic cell (S2, D4) is negative, the implementation of the corresponding Loop(s) will definitely decrease the value of Z. Here there are two distinct loops (Loop 1 and Loop 2) with the same I-index -4, the given TP will have two distinct solutions.

Implementing the Loop 1

By implementing the Loop 1, we obtain the Improved Solution-I as shown in Table 11 and the corresponding computed TTC as shown in Table 12.

Table 11: The Improved Solution-I (Optimal Solution) table by the MODA method

Sources	Destinations				
	D1	D2	D3	D4	**Supply**
S1	3	48	14	**24** 2	**24**
S2	**6** 4	**12** 2	30	**6** 10	**24**
S3	36	8	**2** 12	12	**02**
S4	0	0	**1** 0	**14** 0	**15**
Demand	**6**	**12**	**3**	**44**	**320**

Table 12: The Improved Solution-I (Optimal Solution) with the computed TTC

Basic cells in order	Allocated Quantity	UTC in the basic cell	Quantity × UTC
(S1, D4)	24	2	48
(S2, D1)	06	4	24
(S2, D2)	12	2	24
(S2, D4)	06	10	60
(S3, D3)	02	**12**	24
(S4, D3)	01	0	00
(S4, D4)	14	0	00
		Minimum TTC (Z*)	**180**

Implementing the Loop 2

By implementing the Loop 2, we obtain the Improved Solution-II as shown in Table 13 and the associated computed TTC of shown in Table 14.

Table 13: The Improved Solution-II (Alternative Optimal Solution) by the MODA method

Sources	Destinations				
	D1	D2	D3	D4	Supply
S1	3	48	14	24 2	24
S2	6 4	12 2	30	6 10	24
S3	36	8	12	2 12	02
S4	0	0	3 0	12 0	15
Demand	6	12	3	44	320

Table 14: The Improved Solution-II (Alternative Optimal Solution) with the computed TTC

Basic cells in order	Allocated Quantity	UTC in the basic cell	Quantity × UTC
(S1, D4)	24	2	48
(S2, D1)	06	4	24
(S2, D2)	12	2	24
(S2, D4)	06	10	60
(S3, D4)	02	**12**	24
(S4, D3)	03	0	00
(S4, D4)	12	0	00
		Minimum TTC (Z*)	**180**

Test the optimality of the Improved Solution-I by the MODA method

Step 2: Ensure the Non-degeneracy condition

The obtained solution is non-degenerate as there are exactly 7, that is, (m+n-1) basic cells.

Step 3: Identify the basic cell with the largest UTC

Among the 7 basic cells, the basic cell (S3, D3) is having the largest UTC of $12.

Step 4 & 5: Trace all possible loops starting and ending at the basic cell (S3, D3) and compute its I-index

Trace all the possible loops starting and ending at the identified basic cell (S3, D3) and passing through only one non-basic cell.

Loop 1 = {(S3, D3), (S3, D4), (S4, D4), (S4, D3), (S3, D3)},

NCC = 12-12+0-0 = 0

I-index = **0**

One can verify that from each of the other basic cells, the traced loops will not improve the current solution further. This indicates that the 'optimal level' has reached. Therefore, the Improved Solution-I shown in Table 11 is an optimal solution to the given TP with the minimum TTC of Z* = $180.

Alternative Optimal Solution

It is noted that, at the 'optimal level' the basic cell (S3, D3) is having the I-index zero. This indicates that the given TP has an alternative optimal solution. The alternative optimal solution is obtained by implementing the Loop 1 = {(S3, D3), (S3, D4), (S4, D4), (S4, D3), (S3, D3)} with NCC value 0 on the optimal solution shown on Table 11. Actually, the alternative optimal solution is the Improved Solution-II shown in Table 13 with the same minimum TTC of Z* = $180.

NOTE: If we test the optimality of the Improved Solution-II by the proposed MODA method, we will see that the Improved Solution-II is an optimal solution to the given TP. At the optimal level the basic cell (S3, D4) is having the I-index zero. By implementing the corresponding loop we will see that the Improved Solution-I is an alternative optimal solution. Equivalently, we can derive an alternative optimal solution from a given optimal solution, provided it exists to the given problem.

IMPORTANT NOTE

To reduce the numbers of loops starting and ending at an identified 'basic cell' and passing through only one 'non-basic cell' one can follow the following two restrictions. But they will not work for certain TPs:

Restriction #1

Trace all possible loops starting and ending at the identified basic cell (i, j) with largest UTC C_{ij} and passing through only one non-basic cell (k, l) and through some other basic cells all with UTCs less than or equal to C_{ij}.

Refer Illustrative Example 2, First iteration, Step 4 & 5, Loop 1 is having the non-basic cell (S3, D3) with UTC $12 which is > the UTC $10 of the identified basic cell (S2, D4). Similarly, Loop 2 is having the non-basic cell (S3, D4) with UTC $12 which is > the UTC $10 of the identified basic cell (S2, D4). Each loop is having the NCC value of **-4** and hence each loop is important in improving the current solution. Therefore, one cannot restrict a loop with corner cells having UTCs less than or equal to the UTC C_{ij} of the identified basic cell.

Restriction #2

The one non-basic cell included in a loop must be at an even place (2^{nd}, 4^{th}, 6^{th} etc.) found in the order from the starting basic cell. Recall that a 'loop' in a transportation table is an ordered set of even numbers (≥ 4) of cells having only one non-basic cell and the remaining basic cells.

If one solves the Problem 2 under the UTP column in Table 15, he can see one Loop = {(S2, D3), (S2, D2), **(S3, D2)**, (S3, D1), (S4, D1), (S4, D3), (S2, D3)} with NCC value = **-22** in the 'Second Iteration'. Here the cell (S3, D2) is a non-basic cell at the 3^{rd} place in the order from the identified basic cell (S2, D3). Similarly, he can see another Loop = {(S2, D3), (S4, D3), (S4, D1), (S1, D1), **(S1, D4)**, (S3, D4), (S3, D2), (S4, D2), (S2, D3)} with NCC value = **-11** in the 'Third Iteration'. Here the cell (S1, D4) is a non-basic cell at the 5^{th} place in the order from the identified basic cell (S2, D3). Each loop is important because each is helping to improve the current solution. Therefore, one cannot restrict the place of the non-basic cell in a loop.

4. NUMERICAL EXAMPLES

To validate the efficiency of the proposed MODA method, we have solved a set of 14 numbers of "more challenging" TPs of balanced and unbalanced categories in different small sizes, from various literatures and textbooks, which are listed in Table 15.

Table 15: A set of some "More Challenging" balanced and unbalanced TPs

BTP Problem No.	**UTP Problem No.**
Problem1 $[C_{ij}]$ 3×5= [1 9 13 36 51; 24 1216 20 1; 14 33 1 23 26] $[S_i]$ 3×1= [50, 100, 150] $[D_j]$ 1×5= [100, 70, 50, 40,40]	**Problem 1** $[C_{ij}]$ 3×3= [6 10 14; 12 19 21; 15 14 17] $[S_i]$ 3×1= [50, 50, 50] $[D_j]$ 1×3= [30, 40, 55]
Problem 2 $[C_{ij}]$ 4×5= 4 9 810 12; 6 10 3 2 3; 3 2 7 10 3; 3 5 5 4 8] $[S_i]$ 4×1= [24, 18, 20, 16]$[D_j]$ 1×5= [10, 20, 10, 18, 20]	**Problem 2** $[C_{ij}]$ 3×4= [19 30 50 10; 70 30 40 60; 40 8 70 20] $[S_i]$ 3×1= [7, 9, 18]$[D_j]$ 1×4= [40, 8, 7, 14]
Problem 3 $[C_{ij}]$ 4×6= [1 2 1 4 5 2;3 3 2 1 4 3;4 2 5 9 6 2;3 1 7 3 4 6] $[S_i]$ 4×1= [30, 50, 75, 20] $[D_j]$ 1×6= [20, 40, 30, 10, 50, 25]	**Problem 3** $[C_{ij}]$ 3×4= [10 15 12 12; 8 10 11 9; 11 12 13 10] $[S_i]$ 3×1= [20, 15, 12] $[D_j]$ 1×4= [14, 12, 8, 22]
Problem 4 $[C_{ij}]$ 5×5= [73 40 9 79 20; 62 93 96 8 13; 96 65 80 50 65; 57 58 29 12 87; 56 23 87 18 12] $[S_i]$ 5×1= [8, 7, 9, 3, 5] $[D_j]$ 1×5= [6, 8, 10, 4, 4]	**Problem 4** $[C_{ij}]$ 3×4= [42 48 38 37; 40 49 52 51; 39 38 40 43] $[S_i]$ 3×1= [160, 150, 190] $[D_j]$ 1×4= [80, 90, 110, 160]
Problem 5 $[C_{ij}]$ 5×5= [8 8 2 10 2; 11 4 10 9 4; 5 2 2 11 10; 10 6 6 5 2; 8 11 8 6 4] $[S_i]$ 5×1= [40, 70, 35, 90, 85] $[D_j]$ 1×5= [80, 55, 60, 80, 45]	**Problem 5** $[C_{ij}]$ 4×3= [2 7 14; 3 3 1; 5 4 7; 1 6 2] $[S_i]$ 4×1= [5, 8, 7, 15] $[D_j]$ 1×3= [7, 9, 18]
Problem 6 $[C_{ij}]$ 3×5= [5 7 10 5 3; 8 69 12 14; 10 9 8 10 15] $[S_i]$ 3×1= [5, 10, 10] $[D_j]$ 1×5= [3, 3, 10, 5, 4]	**Problem 6** $[C_{ij}]$ 3×4= [3 48 14 2; 4 230 10; 36 8 12 12] $[S_i]$ 3×1= [24, 24, 2] $[D_j]$ 1×4= [6, 12, 3, 44]
Problem 7 $[C_{ij}]$ 3×4= [6 1 9 3; 11 5 2 8; 10 12 4 7] $[S_i]$ 3×1= [70, 55, 90] $[D_j]$ 1×4= [85, 35, 50, 45]	**Problem 7** $[C_{ij}]$ 3×3= [4 8 8; 16 24 16; 8 16 24] $[S_i]$ 3×1= [76, 82, 77] $[D_j]$ 1×3= [72, 102, 41]

5. RESULT ANALYSIS

For the identified 14 "More Challenging" TPs (7 BTPs and 7 UTPs), as listed in Table 15, the IBFS generated by the I-SOFT method, and the optimal solution derived through the proposed MODA method are shown in Table 16 and Table 17 respectively for the BTPs and UTPs .

Table 16: Derivation of optimal solutions to some "More Challenging" BTPs

BTP #	I-SOFT	Optimal Solution by MODA method	Minimum No. of iterations required to reach the optimal
1.	2810	2700	1
2.	332	316	2
3.	440	430	1
4.	1103	1102	1
5.	1640	1475	2
6.	187	183	1
7. *	1165	1160	1

Table 17: Derivation of optimal solutions to some "More Challenging" UTPs

UTP #	I-SOFT	Optimal Solution by MODA method	Minimum No. of iterations required to reach the optimal
1.	1695	1650	2
2.	883	743	3
3.	4780	4720	1
4.	17060	17050	1
5.	93	75	1
6. *	188	180	1
7. *	2752	2424	1

NOTE: The problems marked with * have alternative optimal solutions.

DECISION

One can easily verify that the MODI method also takes the same minimum numbers of iterations, as shown in Column-4 of Table 16 and Table 17, to reach the optimal solutions. Thereby, the MODA method is an alternative to the MODI method to test the optimality of a solution and also optimizing a solution, if it's not optimal.

ADVANTAGE OF THE PROPOSED 'MODA' METHOD OVER THE EXISTING 'MODI' METHOD

In MODI method, one has to compute the net evaluation (opportunity cost) corresponding to each of the non-basic cells in order to identify the appropriate non-basic cell to enter into a basic one and thereby identify a basic cell to leave as a non-basic cell. But in the proposed MODA method, it is enough to test a very few of the basic cells only to identify the appropriate basic cell to leave as a non-basic one and thereby identify a non-basic cell to enter as a basic cell. Due to this the computational time is considerably less in the proposed MODA method.

6. CONCLUSION

In this paper, we have proposed an innovative iterative technique titled MODA for optimality testing and optimizing a solution found by any available method in transportation problems. The MODA method has been tested on a number of near optimal solutions obtained by the I-SOFT method, the best one for finding the best IBFS to TPs. Testing outcomes authenticate that the proposed MODA method is the perfect one for testing the optimality of an obtained solution and also optimizing of that solution, if that's not optimal. Thereby, the MODA method generates 'tested optimal' solutions to transportation problems. Also, the proposed method generates the alternative optimal solution(s) of a given transportation problem, it they exist to the problem. It is the added benefit of this proposed method. Also, the proposed MODA method is an alternative to the existing MODI method with considerably less computation time than the MODI method.

REFERENCES

[1] Abdul Quddoos, Shakeel, and Khalid M.M., 2012, *A New Method for Finding an Optimal Solution for Transportation Problems,* International Journal on Computer Science and Engineering (IJCSE), 4(7), pp. 1271-1274.

[2] Abdul Quddoos, Shakeel, and Khalid M.M., 2016, *A Revised Version of ASM method for Solving Transportation Problem,* International Journal of Agricult. Stat. Sci., 12,(Supplement 1), pp. 267-272.

[3] Esakkiammal T., Murugesan R., 2021, *SOFTMIN – An Innovative and Efficient Zeros Allocation Method to Solve Transportation Problems*, 1st International Conference on Mathematical Modeling and its Soft Computing Applications (ICMMSCA-2021, November 29-30, 2021, Puttaparthi, Andhra Pradesh, India, Paper ID C-11/ISBN 978-81-952716-2-7.

[4] Kanti Swarup, Gupta P.K. and Man Mohan, 2017, *Operations Research,* 19th Edition, Sultan Chand & Sons, Educational Publishers, New Delhi.

[5] Murugesan R., 2022, *NILA – An Innovative Technique for Optimality Testing and Optimizing a Solution in Assignment Problems,* Global Journal of Pure and

Applied Mathematics, Volume 18, Number 2, pp. 433-452.

[6] Murugesan R., 2022, *SOFTMIN Method versus IASM Method on Some Challenging Transportation Problems*, Global Journal of Pure and Applied Mathematics, Volume 18, Number 1, pp. 341-351.

[7] Murugesan R., 2022, *I-SOFT – The Best Method for Finding the Best IBFS to Transportation Problems*, Global Journal of Pure and Applied Mathematics, Volume 18, Number 01, pp. 377-391.

[8] Murugesan R., 2022, *NEEYA? NAANA? – Some Recent Direct Methods versus Some Challenging Problems in Transportation Problems*, International Journal of Advanced Research, Vol.10, No.05, pp. 526-534.

[9] Murugesan R., Esakkiammal T., 2020, *An Improved ASM Method for the Transportation Problem*, Advances in Mathematics: Scientific Journal, 9(10**),** pp.8259–8271.

[10] Sharma J.K., 2017, *Operations Research-Theory and applications*, Trinity Press, New Delhi.