Available online at https://ganitjournal.bdmathsociety.org/

GANIT: Journal of Bangladesh Mathematical Society

**GANIT** J. Bangladesh Math. Soc. 43.2 (2023) 37–48
DOI: https://doi.org/10.3329/ganit.v43i2.70797.

# A Comparative Study on the Higher-Dimensional Transportation Problems: FSTP and MODI

Nahid Sultana[*a], H.S. Faruque Alam[b], Ganesh Chandra Ray [c],

[a]*Department of Business Administration, International Islamic University Chittagong, Chattogram - 4314, Bangladesh*

[b]*Department of Mathematics, University of Chittagong, Chattogram - 4331, Bangladesh*

[c]*Department of Mathematics, University of Chittagong, Chattogram - 4331, Bangladesh*

## ABSTRACT

An optimal allocation aids a company to get its desired outcome. Their aims are distributed into two core sections; they want to maximize the profit and also try to minimize the related cost. Transportation cost is one of the unwanted costs for the companies. They want to abate it as well. To cut it down, there are a lot of solving methods developed recently. From the recent developments we choose the two effective methods Faster Strongly Polynomial method (FSTP) and the Modified Distribution method worked on Vogel's Approximation Method (VAM-MODI) to find the best one. On our selected higher-dimensional problems, the findings show us that FSTP is best if we compare the number of steps, but concerning the short execution time, VAM-MODI performs well.



**Received:** May 21, 2023 **Accepted:** August 23, 2023 **Published Online:** December 31, 2023

**Keywords:** Higher-dimensional Transportation Problem, Vogel's Approximation Method -Modified Distribution method and Faster Strongly Polynomial method.

***AMS Subject Classifications 2020:*** *90B06, 90C08, 90C90*.

## 1. Introduction

The Transportation Problem (TP) concerns a distribution problem. Here the products are transported from several sources (factories) to several destinations (warehouses) with a cost cut by an optimal allocation. Its main objective is to cut the least cost in transporting these products. There are two restrictions, first, one is the total demand of warehouses and the second one is the total capacity of supplying the products. The transportation problem is classified into different groups based on their primary objective and source supply against destination requirements. With the primary objective, the transportation problem is categorized in two ways, the minimization case and the maximization case. The minimization transportation problem is the case of shipment of commodities where the main goal is to minimize the transportation cost. If a company wants to maximize its earnings by the delivery of the products from sources to destinations, is the maximization case. Considering the source's fixed capacity and the warehouse's fixed demand, there

[*]Corresponding Author. *Email Address:* nstuhin2011@gmail.com

will be two types of TP. Where the supplied quantity and the demanded quantity coincide with each other is called the balanced transportation problem. The opposite in this case is the unbalanced transportation problem. Hence two instances can arise. The number of supply quantities of sources is more than the number of demands of destinations, or vice versa can happen.

The extensions of the transportation problem model help to obtain an optimal solution in the other sectors of the business problem, specifically in employee scheduling, inventory control, personnel assignments, and multi-objective optimization as goal programming. We have passed more than two centuries in finding the algorithm that can provide an optimal solution without the initial basic feasible solution. Hence Vogel's Approximation Method (VAM) and MOdified Distribution Method (MODI) or Stepping Stone Method (SSM) is known as the most efficient method in finding the optimal solution that can satisfy all the constraints and minimize the transportation cost. But in the real world, any transportation problem needs an algorithm that can help the decision-maker to know the optimized result without any computational complexity and in a short computational time. Kleinschmidt's and Schannath (1995) developed a model named STrongly Polynomial (STP) which can give the optimum result without any Initial Basic Feasible Solution (IBFS), but its limitations are that it cuts a considerable run time. In 2018, an algorithm called Faster STrongly Polynomial (FSTP) claimed that it could overcome all the flaws mentioned above. This method motivates us to check if it works better than others or not. We make a comparison of the methods and try to find out which can give us an accurate result in a least steps by cutting the minimum execution time.

## 2. Literature Review

Approximately two centuries ago, we began to find the optimal solution for the problem of transport. According to the literature, the first person who mainly developed the transportation problem was F.L. Hitchcock. He presented his study entitled "The Distribution of a product from several sources to numerous localities''. In 1947, T. C. Koopmans introduced "Optimizing the Use of the Transportation System". Then the transportation problem was converted into a linear programming problem and resolved using the simplex method by the renowned researcher G.B Danzig in 1951. He proposed a MOdified DIstribution method known as MODI to find an initial basic feasible solution in 1963. Charles and Cooper (1954) developed another method named the Stepping Stone Method (SSM) that provides an alternative way of determining the simplex method information. Gleyzal designed an alternative approach in 1955, by Ford and Fulkerson (1955, 1956), and Munkres (1957). It is required to find an initial basic feasible solution to obtain an optimal solution to a transportation problem. In research, many methods are available to achieve an initial basic feasible solution such as the North-West corner rule, Row Minima Method, Column Minima Method, Least Cost Method, Vogel's Approximation method, etc. Reinfeld and Vogel developed the Vogel's Approximation Method, which is usually named VAM or Unit Penalty Method. Some well-known transportation methods include the Stepping Stone Method (Charles and Copper-1954), MOdified DIstribution method (Dantzig, 1963), Modified Stepping Stone method (Shih, 1987), and simplex type algorithm Arsham and Khan, 1989) are used in finding the optimal solution. Further then, many ways were improved by many researchers. Edward J. Russell (1969) proposed Russel's Approximation method where the penalties are calculated by the difference between the corresponding row and column highest entry of every cell from the corresponding element. Then he makes his allocation to having the lowest penalty.

Shimshak et. al. (1981) suggested a modification of VAM for solving the unbalanced problem. Here they followed the VAM as usual by ignoring all the penalties included in the dummy row or column. In 1984 Goyal proposed a method for solving the unbalanced problem where he set the high cost as the dummy cost instead of zero and followed the same procedure as VAM. Ramakrishnan suggested subtracting the smallest element from every row or column and then replacing the dummy cost with the highest unit transportation cost. And VAM is used here for finding the initial basic feasible solution. He developed the GVAM in 1988. Kirca and Satir (1990) concerted the transportation cost matrix. For the Row Opportunity Cost Matrix (ROCM), they subtracted all the lowest values from every element row-wise. For the Column Opportunity Cost Matrix (COCM), they follow the subtraction of all the lowest values from every element's column-wise. Then adding the ROCM and COCM got the Total Opportunity Cost Matrix (TOCM) and used the least cost method to generate a feasible solution. Nagraj Balkrishnan (1990) computed all the column penalties as before, except for the dummy column and the rows, hence the penalties are the difference of the lowest, and the next lowest cost ignoring the dummy column and used as usual VAM. It was discussed in his research "Modified Vogel's Approximation Method for the Unbalanced Transportation Problem". Kore and Thakur (2000) solved the unbalanced transportation problem without converting it to a balanced one.

Ping and Chu (2002) improved the Dual Matrix approach as an alternative to the Stepping Stone by converting the problem into a corresponding dual one using sequence matrix operations. Mathirajan and Meenakshi (2004) modified the procedure followed by Kirca and Satir and defined the penalty of the lowest and 2nd lowest in every row and column and allocation is preferred to the highest penalty cost with a minimum cost cell. Kasana and Kumar (2005) imposed the Extreme Difference Method where VAM is applied to the penalty of the highest and lowest unit transportation cost. Kulkarni and Dattar (2010) converted an unbalanced problem to a balanced one by increasing the demand /supply and proposed a new algorithm to solve it. Abdur Rashid (2011) applied an effective way of finding the initial feasible solution by finding the highest penalty where the penalty is the difference between the extreme and 2nd extreme of each row and column. Mansi (2011) investigates the two alternative methods for solving transportation problems. Mansi Suryakand Gaglani (2011) allocated in the single cell that is the minimum cost point of every row of the cost matrix. If the minimum cost is the same, she breaks the tie by calculating the difference between the minimum, and the next to the minimum unit cost for all those sources where destinations are identical. Aminur Rahman Khan (2011, 2012) calculated the highest cost difference as the penalty of the two highest costs and allocated this way in the most upper penalty with the lowest cost. Sudhakar (2012) developed a new direction in searching for the optimal solution by assigning one zero in each row or column by subtracting the least one from each column and row. Got a suffix value for each zero and considered the greatest one for the allocation. Quddoos et. al. (2012) mentioned in their paper “A New Method for Finding an Optimal Solution for Transportation Problem” that the allocation is preferred to the cell containing the zero and for that make the zeros in every row and column and count the total number of zero. N. M Deshmukh mentioned in his work named “An Innovative Method for Solving Transportation Problem” in 2012 that allocation will be started by subtracting the minimum odd cost for making the cell zero. The Faster Strongly Polynomial Method (FSTP) was developed by Ashraful Babu in the year 2018. The run time is faster than Kleinschmidt’s STP. A comparative study has shown that FSTP provides the optimal solution shortly than the well-known method VAM-MODI on the lower dimension.

## 3. Objectives of the study

The main objective of this study is to find the best one between the selected effective methods in solving the higher dimensional transportation problems. The performances will be evaluated based on some criteria covering the specific objectives. The specific objectives are extended as:

i) To choose here a sufficiently large number of higher-dimensional transportation problems. The problems will be in both category balanced problem and unbalanced also,

ii) To run the selected algorithms by software for time saving calculation. GNU Octave is used as the solver for these methods, codes are generated on GNU Octave and the problems with higher dimensions are solved by it. iii) To measure the performances on account of

a) Optimal solution
b) Number of iterations
c) Execution time

The performance will be measured on the query: finding the best method able to give the optimal solution at least runtime.

iv) To record the performance of the methods. A comparison line chart can show the optimal solution obtained by the two effective methods.

## 4. Methodology

In completing the research work, we have gone through related literature, and we have achieved knowledge in solving the Transportation Problems by various existing algorithms. We have selected two effective methods FSTP and VAM-MODI for higher dimensional problem, as in the lower dimensional problem these two can give us the optimal solution mostly in short time than the other methods including North West Corner Method (NWC), Least Cost Method (LCM), Row Minima Method (RMM), Column Minima Method (CMM), Vogel’s Approximation Method (VAM), Extreme Difference Method (EDM), ASM method (ASM), Revised Distribution Method (RDM), Average Cost Method (ACM), Zero Assignment Method (ZAM), Highest Cost Difference Method (HCDM), Russel’s Approximation Method (RAM), Least Cost Position Method (LCPM), Cost Minimization Approach(CMA), Improved NMD Method (INMD), MTCM-HCDM, TOCM-LCM Approach, TOCM-VAM Approach, TOCM-EDM Approach, TOCM-HCDM Approach, TOCM-SUM Approach and Faster Strongly Polynomial method (FSTP). The optimality test describes the feasible allocations to convert to the optimal allocations. We test the optimality by using one of the most popular methods, the MODI or u-v method where the loop of distribution is restructured. We have worked on 160 problems: 80’s balanced and 80’s unbalanced, and check the solution, no. of iteration and the execution time with the help of GNU Octave. These computer programs are coded by C Programming Language and run on a laptop with Intel Core i3 8GB of RAM. For solving the problem by the methods, we need the software required input. There are three .dat files included here; these are c.dat, demand.dat, and supply.dat. The c.dat file includes the transportation unit matrix, demand.dat includes the demand of the destinations by a column matrix where the 1 d shows the demand of the 1st destination, 2 d represents the demand of the 2nd destination, and so on. The supply.dat file presents the capacity of each source or factory. This file also inputs the data in a column matrix where 1 s shows the demand of the 1st source, 2 s represents the demand of the 2nd source, and so on. Now we have compared the effective method to MODI, the most popular method in the optimality test. We consider the criterion of the best method in obtaining the optimal solution.

## 5. Statement of the problem

The transportation problem mainly evaluated the quantity of distributed products from the different sources to the different destinations. There will be at least 2 sources and 2 destinations. A manufacturing company has $m^{th}$ plants to produce their product. They have $n^{th}$ warehouses to distribute their product. The unit cost of delivering the products from the plant $P_1$, $P_2$,.…..and $P_m$ to the warehouses $D_1$, $D_2$, $D_3$,.......and $D_n$ are taka $c_{11}$, $c_{12}$,$c_{13}$,......$c_{1n}$; $c_{21}$, $c_{22}$, $c_{23}$,......$c_{2n}$;..........; $c_{m1}$, $c_{m2}$, $c_{m3}$,......$c_{mn}$ respectively. The demands of the warehouses are $b_1$, $b_2$, $b_3$, …...and $b_n$ units respectively. The capacity of producing the products are $a_1$, $a_2$, $a_3$, …...and $a_m$ units respectively. The company should know the optimal quantity of delivering the products from the plants to the warehouses that will be helpful to cut a minimum cost. The transportation problem is given in the tabular form:

Table 1: General Transportation Problem

| | $D_1$ | $D_2$ | $D_3$ | …. | $D_4$ | Supply |
|---|---|---|---|---|---|---|
| $P_1$ | $c_{11}$ | $c_{12}$ | $c_{13}$ | …. | $c_{1n}$ | $a_1$ |
| $P_2$ | $c_{21}$ | $c_{22}$ | $c_{23}$ | …. | $c_{2n}$ | $a_2$ |
| $P_3$ | $c_{31}$ | $c_{32}$ | $c_{33}$ | …. | $c_{3n}$ | $a_3$ |
| ….. | …. | …. | …. | …. | …. | …. |
| $P_m$ | $c_{m1}$ | $c_{m2}$ | $c_{m3}$ | …. | $c_{mn}$ | $a_m$ |
| Demand | $b_1$ | $b_2$ | $b_3$ | …. | $b_n$ | |

A transportation problem is balanced if the total supply ($a_i$) from all sources is equal to the total demand ($b_j$) in the destinations i.e.,

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$$

A transportation problem is said to be unbalanced if the total supply ($a_i$) from all sources is not equal to the total demand ($b_j$) in the destinations i.e.,

$$\sum_{i=1}^{m} a_i \neq \sum_{j=1}^{n} b_j$$

The mathematical formulation of the above general transportation problem is [26]

$$Z_{min} = \sum_{i=1}^{m}\sum_{j=1}^{n} C_{ij} x_{ij}$$

Subject to,

$$\sum_{j=1}^{n} x_{ij} = a_i, i = 1,2,3,...,m$$

$$\sum_{i=1}^{m} x_{ij} = b_j, j = 1,2,3,...,n$$

$$where, x_{ij} \geq 0$$

## 6. Model Development

FSTP is an improvement of the STP method. It hangs the maximum cost firstly, and it does not delete any satisfied row or column that helps us to allow the reallocation when it needs. A better allocation can be made at any step for any cell. There is no need for an initial basic feasible solution. It is easy to calculate by this procedure causes its target one cell first, and it computes only row-wise or column-wise. This method targets the algorithm of the optimal testing method MODI. On our observation, out of randomly selected 160 problems, all problems can be solved with the optimal solution by this method. It provides the same solution as MODI, and it can reduce the number of steps, i.e. iterations. But it takes a little bit a long time to execute. Based on the survey, it is found that the probability of getting the optimal solution is about 100%.

## 7. Findings

Now, we will check which method between MODI and FSTP provides us with an optimal solution with a lower number of iterations and most moderate execution time in the case of higher dimensions

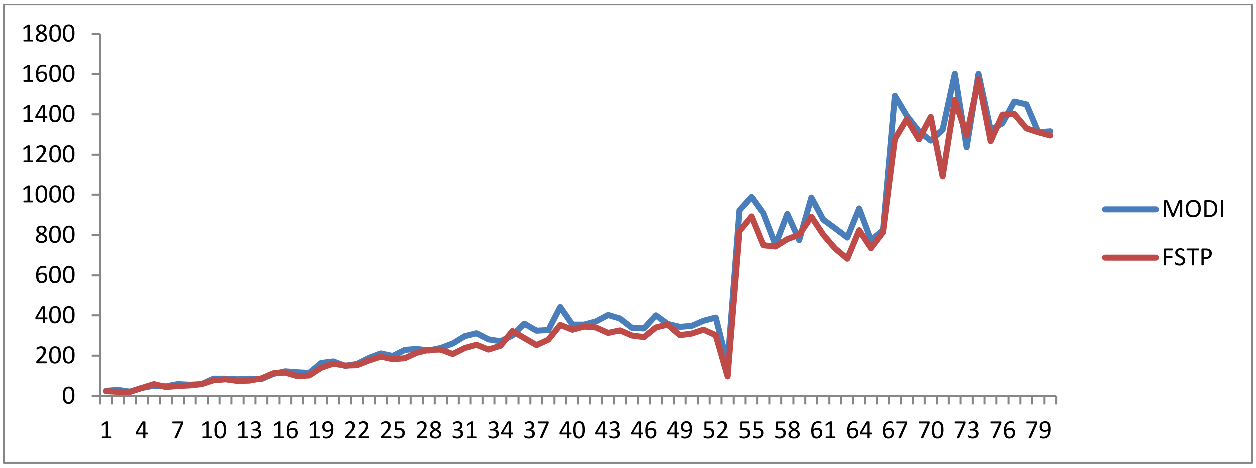


***Figure 1: Comparison based on the no. of iterations between MODI and FSTP (Balanced Problems)***

***Source: Table 1 (Appendix) (Table of the No. of Iterations, in selected methods)***

From 80 randomly selected cases, 11's has a larger number of iterations for FSTP, 2's have the same number of steps, and the remaining 67 problems can reduce the total number of steps by FSTP. That means from the whole selected balanced cases, 84% can reduce the levels while 14% cannot, and 3% show the same.

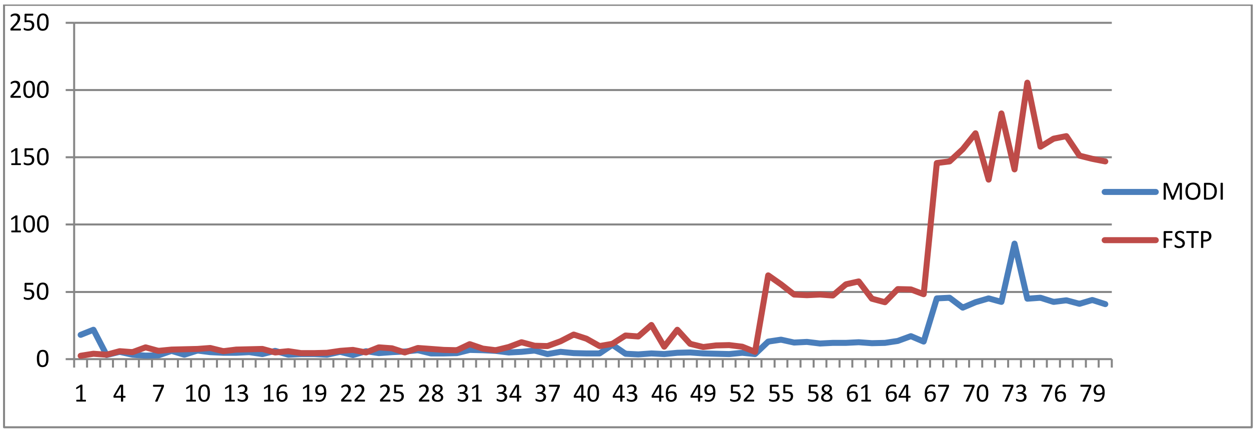


***Figure 2: Comparison based on the execution time between MODI and FSTP (Balanced Problems)***

***Source: Table 2 (Appendix) (Table of Execution time in solving by selected methods)***

FSTP considers the processing time include the starting of data entry. From the above figure, it is clear that the FSTP method takes a long time in comparison with MODI. The figure also shows the higher dimension takes a higher difference in execution time. Here the fluctuation is for solving the critical cases. There are a few cases where these two methods take the same execution time.

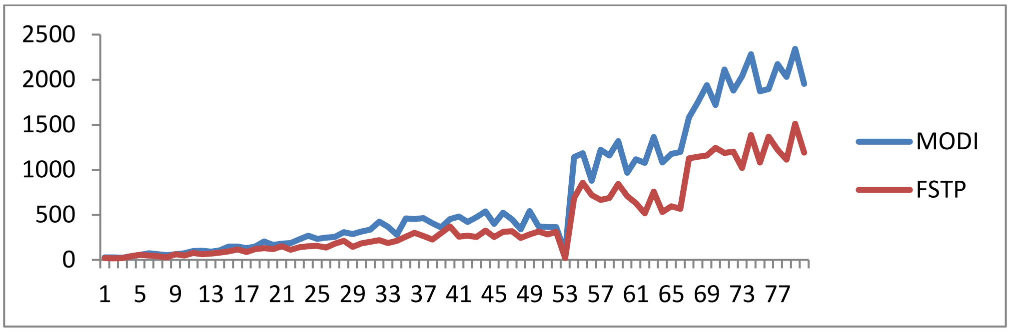


***Figure 3: Comparison based on the no. of iterations between MODI and FSTP (Unbalanced Problems)***

***Source: Table 3 (Appendix) (Table of the no. of iteration by selected methods)***

Here 80 unbalanced cases are chosen for checking the effectiveness of the two effective methods. The higher dimensional case has been selected for this comparison. The comparison table for the unbalanced problem shown that, merely 97.5% of cases can reduce the number of steps by the FSTP method. Another 2.5% has the same amount of levels. MODI needs more steps to solve the TP problems as it needs IFS to work on.

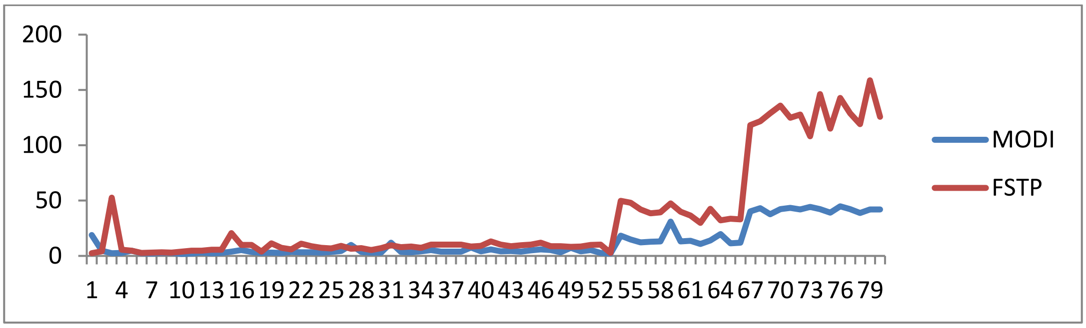


***Figure 4: Comparison based on the execution time between MODI and FSTP (Unbalanced Problems)***

***Source: Table 4 (Appendix) (Table of Execution time in solving by selected methods)***

FSTP considers the processing time include the starting of data entry. From the above figure, it is clear that the FSTP method takes a long time in comparison with MODI. The figure also shows the higher dimension takes a higher difference in execution time for FSTP.

In the sense of the number of steps, FSTP is better than MODI because MODI can only work on the initial basic feasible solution. But FSTP can do the same work without the initial basic feasible solution. But if the consideration goes to the execution time, MODI can provide a much faster result than FSTP.

## 8. Conclusions

FSTP is the most effective method of finding the optimal solution comparing the VAM-MODI. The important advantage of this method is there is no need of finding the IBFS (Initial Basic Feasible Solution). With this technique, it is not necessary to add any row or column to make an unbalanced one into a balanced one. The comparative study shows that FSTP gives all the optimal solutions. It is a faster method in the sense of the number of iterations. On the other hand, MODI (Modified Distribution Method) is another way to find the optimal solution. As Stepping Stone Method, this MODI method is also used in the optimality test and is a very easy way to check the optimality. But the main problem is, it can work on only the initial feasible solution, and between the existing methods except for FSTP, Vogel's Approximation Method gives the more appropriate solution. Hence the FSTP provides the optimal solution for all our randomly selected different dimensional 160 problems, with a comparatively a smaller number of iterations but cutting a comparative large execution time. And obviously, this does not mean that there is no chance of losing the optimal solution by using this method, it may be but we have got the optimal solution by using this method in our study. There are some limitations, one of these is our data selection, here we just consider the linear transportation minimization problem, the dimensions of more than 300×300 are not tested here, and GNU Octave may have some technical flaws. Further research can be continued by increasing the dimension of the complex problem. Anyone can work in developing to fasten this FSTP method, i.e., which helps it to cut the least execution time.

**References**

[1] Adlakha, V., & Kowalski, K., Alternate Solutions Analysis for Transportation Problems, Journal of Business & Economic Research, 7 (11) (2009), 41–49.

[2] Ahmed, M. M., Khan, A. R., Uddin, M. S. & Ahmed, F., A New Approach to Solve Transportation Problems, Open Journal of Optimization, 5 (2016), 22–30.

[3] Amirul, I., Khan, A.R., Uddin, M.S., & Malek, M.A.,Determination of Basic Feasible Solution of Transportation Problem: A New Approach, Jahangirnagar University Journal of Science, 35(1) (2012), 105-112.

[4] Babu, M. A., An Alternative Optimal Solution Algorithm to the Transportation Problem, Ph.D. thesis, Department of Mathematics, Jahangirnagar University,2018.

[5] Balakrishnan, N., Modified Vogel's Approximation Method for Unbalanced Transportation Problem, Applied Mathematics Letters, 3(2) (1990), 9-11.

[6] Charnes, A., Cooper, W. W., & Henderson, An Introduction to Linear programming (Wiley, New York), 1953.

[7] Dantzig, G. B., Application of the Simplex Method to a Transportation problem, Activity Analysis of Production and Allocation, New York–London 1951 (Wiley & Chapman–Hall),(1947),359–373.

[8] Elias, M.H., & Ray G. C., Linear Programming, Fourth Edition, Titas Publications, Bangladesh. 2009.

[9] Fredrick, S.H., & Lieberman, G. J., Operations Research, TATA McGraw Hill Education Pvt. Ltd., New Delhi, 2009.

[10] Gass, S.I., On Solving the Transportation Problem, Journal of Operational Research Society, 35(4), (1990), 291-297.

[11] Gupta, P.K. & Mahan, M. , Problems in Operations Research, Sultan Chand& Company Ltd, New Delhi, 2003.

[12] Gupta, P.K. & Hira, D.S., Operations Research, Sultan Chand & Company Ltd, New Delhi, 2009.

[13] Hitchcock, F. L., The Distribution of a Product from Several Sources to Numerous Localities, Journal of Mathematics and Physics, 20(1941), 224–230.

[14] Kapoor, V.K. ,Operations Research, Sultan Chand& Company Ltd, New Delhi, 2002.

[15] Kirca, O., & Satir, A., A Heuristic for Obtaining an Initial Solution for the Transportation Problem, Journal of Operational Research Society, 41(9) (1990), 865-871.

[16] Kleinschmidt, P., & Schannath, H., A Strongly Polynomial Algorithm for the Transportation Problem, Mathematical Programming, 68(1995), 1–13.

[17] Koopmans, T. C., Optimum Utilization of Transportation System, Econometrica [Supplement], 17(1949), 136–146.

[18] Kore B.G., & Thakur L.B., A New Method of Obtaining Optimal Solution to Unbalanced Transportation Problem, Journal of India Academy of Mathematics, 22(2) (2000), 247-258.

[19] Korukoğlu, S., & Balli, S., An Improved Vogel's Approximation Method for the Transportation Problem, Mathematical & Computational Applications, 16 (2) (2011), 370–381.

[20] Korukoglu, S., & Balli, S. ,An Improved Vogel's Approximation Method for the Transportation Problem, Association for Scientific research, Mathematical and Computational Application, 16(2) (2011), 370-381.

[21] Orden, A., The Transshipment Problem, Management Science, 2 (3) (1956), 276–285.

[22] Orlin, J. B., A Faster Strongly Polynomial Minimum Cost Flow Algorithm. Proceedings 20th ACM Symposium on the Theory of Computation, 1988.

[23] Reinfeld, N. V., & Vogel. W. R., Mathematical Programming, Englewood Cliffs, N: Prentice-Hall, 1958.

[24] Schrijver, A., On the History of the Transportation and Maximum Flow Problems, Mathematical Programming, 91(2002), 437–445.

[25] Shimshak, D. G., Kaslik, J. A., & Barclay, T. D., A Modification of Vogel's Approximation Method through the Use of Heuristic. Information System and Operational Research ( INFOR), 19(1981), 259–263.

[26] Soomro, A. S., Junaid, M., & Tularam, G. A., Modified Vogel's Approximation Method for Solving Transportation Problems. Mathematical Theory and Modeling, 5 (4) (2015), 32–42.

[27] Sultana, N., Alam, H. F., & Ray, G. C., A Comparison Study between the Recently Developed Methods of Transportation Problem:A Study on the Lower-Dimensional Problems. GANIT: Journal of Bangladesh Mathematical Society, 42(1) (2022), 035–049.

[28] Ullah, M. W., Uddin, M. A., & Kawser, R., A Modified Vogel's Approximation Method for Obtaining a Good Primal Solution of Transportation Problems, Annals of Pure and Applied Mathematics, 11 (1) (2016), 63–71.

[29] Vaidyanathan, B., Faster Strongly Polynomial Algorithms for the Unbalanced Transportation Problem and Assignment Problem with Monge costs, Networks, 62(2013), 136–148.

[30] Vasko, F. J., & Storozhyshina, N., Balancing a Transportation Problem: Is it Really that Simple? OR Insight, 24 (3) (2011), 205–214.

Appendix

**Comparison of Randomly selected Balanced Problem**

Table 1: Comparison Table of balanced problems based on optimal solution, No of Iterations, Execution time (cont.)

| Method No. | | ***Optimal Sol.*** | Total Cost | | | No of Iteration | | | Execution time (Sec.) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Case | Size | | MODI | FSTP | Diff. | MODI | FSTP | Diff. | MODI | FSTP | Diff. |
| 41 | 10x10 | 12377 | 12377 | 12377 | 0 | 26 | 23 | -3 | 18.128 | 2.485 | -15.643 |
| 42 | 10x10 | 104310 | 104310 | 104310 | 0 | 30 | 21 | -9 | 21.952 | 4.076 | -17.876 |
| 43 | 10x10 | 80566 | 80566 | 80566 | 0 | 21 | 20 | -1 | 3.1 | 3.305 | 0.205 |
| 44 | 10x20 | 205613 | 205613 | 205613 | 0 | 39 | 39 | 0 | 5.407 | 5.817 | 0.41 |
| 45 | 10x30 | 587430 | 587430 | 587430 | 0 | 51 | 58 | 7 | 3.373 | 5.28 | 1.907 |
| 46 | 20x20 | 108600 | 108600 | 108600 | 0 | 47 | 44 | -3 | 2.612 | 8.793 | 6.181 |
| 47 | 20x20 | 131363 | 131363 | 131363 | 0 | 58 | 49 | -9 | 2.835 | 6.243 | 3.408 |
| 48 | 20x20 | 200903 | 200903 | 200903 | 0 | 55 | 53 | -2 | 6.253 | 7.102 | 0.849 |
| 49 | 20x20 | 238969 | 238969 | 238969 | 0 | 58 | 59 | 1 | 3.229 | 7.335 | 4.106 |
| 50 | 20x30 | 198873 | 198873 | 198873 | 0 | 86 | 77 | -9 | 6.47 | 7.566 | 1.096 |
| 51 | 30x30 | 156542 | 156542 | 156542 | 0 | 86 | 82 | -4 | 5.248 | 8.358 | 3.11 |
| 52 | 30x30 | 155026 | 155026 | 155026 | 0 | 83 | 74 | -9 | 4.676 | 6.004 | 1.328 |
| 53 | 30x30 | 179137 | 179137 | 179137 | 0 | 85 | 76 | -9 | 4.796 | 7.126 | 2.33 |
| 54 | 30x30 | 144849 | 144849 | 144849 | 0 | 84 | 88 | 4 | 5.127 | 7.364 | 2.237 |
| 55 | 40x40 | 180506 | 180506 | 180506 | 0 | 110 | 113 | 3 | 3.668 | 7.624 | 3.956 |
| 56 | 40x40 | 191153 | 191153 | 191153 | 0 | 122 | 116 | -6 | 6.094 | 4.87 | -1.224 |
| 57 | 40x40 | 158762 | 158762 | 158762 | 0 | 117 | 98 | -19 | 3.35 | 5.832 | 2.482 |
| 58 | 40x40 | 146310 | 146310 | 146310 | 0 | 114 | 101 | -13 | 3.741 | 4.361 | 0.62 |
| 59 | 50x50 | 161584 | 161584 | 161584 | 0 | 164 | 139 | -25 | 3.852 | 4.443 | 0.591 |
| 60 | 50x50 | 230225 | 230225 | 230225 | 0 | 172 | 160 | -12 | 3.303 | 4.683 | 1.38 |
| 61 | 50x50 | 189692 | 189692 | 189692 | 0 | 149 | 151 | 2 | 5.388 | 6.199 | 0.811 |
| 62 | 50x50 | 207311 | 207311 | 207311 | 0 | 159 | 153 | -6 | 3.032 | 6.878 | 3.846 |
| 63 | 60x60 | 189903 | 189903 | 189903 | 0 | 189 | 176 | -13 | 5.839 | 5.036 | -0.803 |
| 64 | 60x60 | 206988 | 206988 | 206988 | 0 | 212 | 196 | -16 | 4.502 | 8.802 | 4.3 |
| 65 | 60x60 | 226871 | 226871 | 226871 | 0 | 199 | 182 | -17 | 5.178 | 7.987 | 2.809 |
| 66 | 60x60 | 241630 | 241630 | 241630 | 0 | 229 | 188 | -41 | 5.769 | 4.974 | -0.795 |
| 67 | 70x70 | 217494 | 217494 | 217494 | 0 | 234 | 214 | -20 | 6.497 | 8.367 | 1.87 |
| 68 | 70x70 | 212612 | 212612 | 212612 | 0 | 226 | 229 | 3 | 4.123 | 7.538 | 3.415 |
| 69 | 70x70 | 209226 | 209226 | 209226 | 0 | 239 | 230 | -9 | 4.136 | 6.771 | 2.635 |
| 70 | 70x70 | 248785 | 248785 | 248785 | 0 | 261 | 209 | -52 | 4.474 | 6.636 | 2.162 |
| 71 | 80x80 | 218265 | 218265 | 218265 | 0 | 297 | 238 | -59 | 6.796 | 11.241 | 4.445 |
| 72 | 80x80 | 231035 | 231035 | 231035 | 0 | 312 | 255 | -57 | 6.585 | 7.783 | 1.198 |
| 73 | 80x80 | 211462 | 211462 | 211462 | 0 | 281 | 230 | -51 | 6.231 | 6.726 | 0.495 |
| 74 | 80x80 | 218245 | 218245 | 218245 | 0 | 272 | 249 | -23 | 5.029 | 8.967 | 3.938 |
| 75 | 90x90 | 218892 | 218892 | 218892 | 0 | 300 | 323 | 23 | 5.381 | 12.586 | 7.205 |
| 76 | 90x90 | 284981 | 284981 | 284981 | 0 | 359 | 287 | -72 | 6.312 | 9.876 | 3.564 |
| 77 | 90x90 | 206425 | 206425 | 206425 | 0 | 324 | 253 | -71 | 3.695 | 9.623 | 5.928 |
| 78 | 90x90 | 233036 | 233036 | 233036 | 0 | 328 | 280 | -48 | 5.52 | 13.341 | 7.821 |
| 79 | 100x100 | 431390 | 431390 | 431390 | 0 | 442 | 353 | -89 | 4.418 | 18.217 | 13.799 |
| 80 | 100x100 | 254952 | 254952 | 254952 | 0 | 354 | 329 | -25 | 4.305 | 15.109 | 10.804 |

Table 2: Comparison Table of balanced problems based on optimal solution, No of Iterations, Execution time

| Method No. | | ***Optimal Sol.*** | Total Cost | | | No of Iteration | | | Execution time (Sec.) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Case | Size | | MODI | FSTP | Diff. | MODI | FSTP | Diff. | MODI | FSTP | Diff. |
| 81 | 100x100 | 269963 | 269963 | 269963 | 0 | 355 | 345 | -10 | 4.188 | 9.747 | 5.559 |
| 82 | 100x100 | 258227 | 258227 | 258227 | 0 | 370 | 340 | -30 | 10.745 | 11.306 | 0.561 |
| 83 | 100x100 | 260602 | 260602 | 260602 | 0 | 402 | 313 | -89 | 3.911 | 17.449 | 13.538 |
| 84 | 100x100 | 246227 | 246227 | 246227 | 0 | 385 | 326 | -59 | 3.53 | 16.792 | 13.262 |
| 85 | 100x100 | 227633 | 227633 | 227633 | 0 | 338 | 301 | -37 | 4.284 | 25.412 | 21.128 |
| 86 | 100x100 | 239896 | 239896 | 239896 | 0 | 336 | 292 | -44 | 3.867 | 9.129 | 5.262 |
| 87 | 100x100 | 245172 | 245172 | 245172 | 0 | 401 | 340 | -61 | 4.823 | 21.81 | 16.987 |
| 88 | 100x100 | 239896 | 239896 | 239896 | 0 | 357 | 355 | -2 | 4.945 | 11.438 | 6.493 |
| 89 | 100x100 | 204842 | 204842 | 204842 | 0 | 343 | 302 | -41 | 4.24 | 8.993 | 4.753 |
| 90 | 100x100 | 221857 | 221857 | 221857 | 0 | 348 | 310 | -38 | 4.099 | 10.164 | 6.065 |
| 91 | 100x100 | 251888 | 251888 | 251888 | 0 | 373 | 329 | -44 | 3.663 | 10.508 | 6.845 |
| 92 | 100x100 | 23524 | 23524 | 23524 | 0 | 390 | 302 | -88 | 4.713 | 9.216 | 4.503 |
| 93 | 200x200 | 11932 | 11932 | 11932 | 0 | 157 | 97 | -60 | 3.703 | 5.397 | 1.694 |
| 94 | 200x200 | 406512 | 406512 | 406512 | 0 | 923 | 820 | -103 | 13.055 | 62.366 | 49.311 |
| 95 | 200x200 | 419405 | 419405 | 419405 | 0 | 989 | 892 | -97 | 14.472 | 55.515 | 41.043 |
| 96 | 200x200 | 376871 | 376871 | 376871 | 0 | 909 | 750 | -159 | 12.333 | 47.97 | 35.637 |
| 97 | 200x200 | 365457 | 365457 | 365457 | 0 | 751 | 743 | -8 | 12.776 | 47.59 | 34.814 |
| 98 | 200x200 | 382324 | 382324 | 382324 | 0 | 905 | 780 | -125 | 11.52 | 48.084 | 36.564 |
| 99 | 200x200 | 417302 | 417302 | 417302 | 0 | 774 | 802 | 28 | 12.102 | 47.287 | 35.185 |
| 100 | 200x200 | 442786 | 442786 | 442786 | 0 | 986 | 891 | -95 | 12.1 | 55.549 | 43.449 |
| 101 | 200x200 | 368462 | 368462 | 368462 | 0 | 877 | 802 | -75 | 12.49 | 57.772 | 45.282 |
| 102 | 200x200 | 395301 | 395301 | 395301 | 0 | 833 | 734 | -99 | 11.957 | 45.008 | 33.051 |
| 103 | 200x200 | 358471 | 358471 | 358471 | 0 | 788 | 682 | -106 | 12.048 | 42.305 | 30.257 |
| 104 | 200x200 | 428092 | 428092 | 428092 | 0 | 932 | 824 | -108 | 13.578 | 52.105 | 38.527 |
| 105 | 200x200 | 365746 | 365746 | 365746 | 0 | 774 | 735 | -39 | 17.175 | 51.699 | 34.524 |
| 106 | 200x200 | 406230 | 406230 | 406230 | 0 | 824 | 815 | -9 | 12.924 | 48.194 | 35.27 |
| 107 | 300x300 | 540038 | 540038 | 540038 | 0 | 1492 | 1277 | -215 | 45.133 | 145.675 | 100.542 |
| 108 | 300x300 | 502925 | 502925 | 502925 | 0 | 1393 | 1376 | -17 | 45.706 | 146.978 | 101.272 |
| 109 | 300x300 | 528073 | 528073 | 528073 | 0 | 1315 | 1276 | -39 | 38.253 | 155.986 | 117.733 |
| 110 | 300x300 | 500774 | 500774 | 500774 | 0 | 1270 | 1388 | 118 | 42.273 | 167.804 | 125.531 |
| 111 | 300x300 | 502620 | 502620 | 502620 | 0 | 1323 | 1091 | -232 | 45.044 | 133.369 | 88.325 |
| 112 | 300x300 | 590438 | 590438 | 590438 | 0 | 1602 | 1472 | -130 | 42.627 | 182.765 | 140.138 |
| 113 | 300x300 | 517264 | 517264 | 517264 | 0 | 1236 | 1295 | 59 | 85.878 | 141.024 | 55.146 |
| 114 | 300x300 | 583877 | 583877 | 583877 | 0 | 1602 | 1575 | -27 | 44.943 | 205.516 | 160.573 |
| 115 | 300x300 | 489260 | 489260 | 489260 | 0 | 1322 | 1266 | -56 | 45.686 | 158.034 | 112.348 |
| 116 | 300x300 | 520514 | 520514 | 520514 | 0 | 1355 | 1399 | 44 | 42.562 | 163.909 | 121.347 |
| 117 | 300x300 | 483974 | 483974 | 483974 | 0 | 1463 | 1401 | -62 | 43.667 | 165.83 | 122.163 |
| 118 | 300x300 | 575250 | 575250 | 575250 | 0 | 1449 | 1330 | -119 | 41.191 | 151.194 | 110.003 |
| 119 | 300x300 | 534636 | 534636 | 534636 | 0 | 1311 | 1311 | 0 | 43.948 | 148.873 | 104.925 |
| 120 | 300x300 | 478935 | 478935 | 478935 | 0 | 1316 | 1295 | -21 | 40.874 | 146.94 | 106.066 |

**Comparison of Unbalanced Problem**

Table 3: Comparison Table of unbalanced problems based on optimal solution, No of Iterations, Execution time (cont.)

| Method No. | | ***Optimal Sol.*** | Total Cost | | | No of Iterations | | | Execution time (Sec.) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Case | Size | | MODI | FSTP | Diff. | MODI | FSTP | Diff. | MODI | FSTP | Diff. |
| 121 | 10x10 | 7270 | 7270 | 7270 | 0 | 26 | 19 | -7 | 18.937 | 2.466 | -16.471 |
| 122 | 10x10 | 51524 | 51524 | 51524 | 0 | 26 | 17 | -9 | 4.807 | 4.087 | -0.72 |
| 123 | 10x10 | 70399 | 70399 | 70399 | 0 | 23 | 20 | -3 | 2.356 | 52.776 | 50.42 |
| 124 | 10x20 | 291561 | 291561 | 291561 | 0 | 40 | 40 | 0 | 2.829 | 5.445 | 2.616 |
| 125 | 10x30 | 575195 | 575195 | 575195 | 0 | 55 | 55 | 0 | 4.806 | 4.621 | -0.185 |
| 126 | 20x20 | 106321 | 106321 | 106321 | 0 | 74 | 47 | -27 | 2.317 | 2.678 | 0.361 |
| 127 | 20x20 | 86012 | 86012 | 86012 | 0 | 64 | 41 | -23 | 2.066 | 3.023 | 0.957 |
| 128 | 20x20 | 79907 | 79907 | 79907 | 0 | 52 | 28 | -24 | 2.598 | 3.226 | 0.628 |
| 129 | 20x20 | 192607 | 192607 | 192607 | 0 | 64 | 62 | -2 | 2.643 | 2.863 | 0.22 |
| 130 | 20x30 | 90156 | 90156 | 90156 | 0 | 73 | 47 | -26 | 1.744 | 3.718 | 1.974 |
| 131 | 30x30 | 106050 | 106050 | 106050 | 0 | 99 | 75 | -24 | 2.046 | 4.768 | 2.722 |
| 132 | 30x30 | 108635 | 108635 | 108635 | 0 | 100 | 61 | -39 | 3.063 | 4.744 | 1.681 |
| 133 | 30x30 | 134203 | 134203 | 134203 | 0 | 90 | 71 | -19 | 2.545 | 5.698 | 3.153 |
| 134 | 30x30 | 159913 | 159913 | 159913 | 0 | 106 | 82 | -24 | 2.72 | 5.602 | 2.882 |
| 135 | 40x40 | 132385 | 132385 | 132385 | 0 | 146 | 95 | -51 | 3.767 | 20.547 | 16.78 |
| 136 | 40x40 | 190934 | 190934 | 190934 | 0 | 146 | 115 | -31 | 5.171 | 10.063 | 4.892 |
| 137 | 40x40 | 130121 | 130121 | 130121 | 0 | 131 | 89 | -42 | 3.576 | 9.827 | 6.251 |
| 138 | 40x40 | 190787 | 190787 | 190787 | 0 | 149 | 118 | -31 | 2.686 | 3.929 | 1.243 |
| 139 | 50x50 | 165739 | 165739 | 165739 | 0 | 206 | 129 | -77 | 2.913 | 11.429 | 8.516 |
| 140 | 50x50 | 155478 | 155478 | 155478 | 0 | 164 | 119 | -45 | 2.59 | 7.423 | 4.833 |
| 141 | 50x50 | 188072 | 188072 | 188072 | 0 | 180 | 151 | -29 | 3.601 | 5.856 | 2.255 |
| 142 | 50x50 | 158054 | 158054 | 158054 | 0 | 187 | 112 | -75 | 3.166 | 10.965 | 7.799 |
| 143 | 60x60 | 141114 | 141114 | 141114 | 0 | 231 | 141 | -90 | 3.335 | 8.692 | 5.357 |
| 144 | 60x60 | 172394 | 172394 | 172394 | 0 | 269 | 153 | -116 | 2.994 | 7.218 | 4.224 |
| 145 | 60x60 | 143728 | 143728 | 143728 | 0 | 232 | 154 | -78 | 3.48 | 6.703 | 3.223 |
| 146 | 60x60 | 163270 | 163270 | 163270 | 0 | 248 | 138 | -110 | 4.374 | 8.944 | 4.57 |
| 147 | 70x70 | 172114 | 172114 | 172114 | 0 | 255 | 181 | -74 | 9.886 | 6.441 | -3.445 |
| 148 | 70x70 | 199106 | 199106 | 199106 | 0 | 307 | 212 | -95 | 3.686 | 6.923 | 3.237 |
| 149 | 70x70 | 170429 | 170429 | 170429 | 0 | 286 | 145 | -141 | 2.971 | 5.235 | 2.264 |
| 150 | 70x70 | 174618 | 174618 | 174618 | 0 | 314 | 183 | -131 | 3.109 | 6.961 | 3.852 |
| 151 | 80x80 | 185578 | 185578 | 185578 | 0 | 336 | 200 | -136 | 11.941 | 9.508 | -2.433 |
| 152 | 80x80 | 219365 | 219365 | 219365 | 0 | 426 | 219 | -207 | 3.639 | 7.926 | 4.287 |
| 153 | 80x80 | 188719 | 188719 | 188719 | 0 | 367 | 188 | -179 | 3.173 | 8.609 | 5.436 |
| 154 | 80x80 | 187060 | 187060 | 187060 | 0 | 279 | 213 | -66 | 4.084 | 7.365 | 3.281 |
| 155 | 90x90 | 225529 | 225529 | 225529 | 0 | 460 | 259 | -201 | 5.279 | 10.161 | 4.882 |
| 156 | 90x90 | 292424 | 292424 | 292424 | 0 | 453 | 300 | -153 | 3.8 | 10.344 | 6.544 |
| 157 | 90x90 | 234730 | 234730 | 234730 | 0 | 463 | 266 | -197 | 3.903 | 10.209 | 6.306 |
| 158 | 90x90 | 184088 | 184088 | 184088 | 0 | 406 | 227 | -179 | 3.859 | 10.184 | 6.325 |
| 159 | 100x100 | 217658 | 217658 | 217658 | 0 | 362 | 296 | -66 | 7.575 | 8.505 | 0.93 |
| 160 | 100x100 | 334357 | 334357 | 334357 | 0 | 451 | 372 | -79 | 4.211 | 8.914 | 4.703 |

Table 4: Comparison Table of unbalanced problems based on optimal solution, No of Iterations, Execution time

| Method No. | | ***Optimal Sol.*** | Total Cost | | | No of Iteration | | | Execution time (Sec.) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Case | Size | | MODI | FSTP | Diff. | MODI | FSTP | Diff. | MODI | FSTP | Diff. |
| 161 | 100x100 | 211436 | 211436 | 211436 | 0 | 480 | 258 | -222 | 5.961 | 13.146 | 7.185 |
| 162 | 100x100 | 246469 | 246469 | 246469 | 0 | 420 | 270 | -150 | 4.037 | 10.217 | 6.18 |
| 163 | 100x100 | 215501 | 215501 | 215501 | 0 | 473 | 253 | -220 | 4.371 | 8.777 | 4.406 |
| 164 | 100x100 | 229589 | 229589 | 229589 | 0 | 537 | 325 | -212 | 3.891 | 9.688 | 5.797 |
| 165 | 100x100 | 208829 | 208829 | 208829 | 0 | 398 | 255 | -143 | 5.067 | 10.146 | 5.079 |
| 166 | 100x100 | 238032 | 238032 | 238032 | 0 | 524 | 311 | -213 | 6.015 | 11.855 | 5.84 |
| 167 | 100x100 | 244900 | 244900 | 244900 | 0 | 448 | 318 | -130 | 5.349 | 8.797 | 3.448 |
| 168 | 100x100 | 208603 | 208603 | 208603 | 0 | 339 | 242 | -97 | 3.389 | 8.75 | 5.361 |
| 169 | 100x100 | 253010 | 253010 | 253010 | 0 | 542 | 283 | -259 | 7.236 | 8.326 | 1.09 |
| 170 | 100x100 | 267969 | 267969 | 267969 | 0 | 370 | 314 | -56 | 4.033 | 8.566 | 4.533 |
| 171 | 100x100 | 223476 | 223476 | 223476 | 0 | 365 | 284 | -81 | 5.373 | 9.832 | 4.459 |
| 172 | 100x100 | 24086 | 24086 | 24086 | 0 | 363 | 310 | -53 | 2.831 | 10.107 | 7.276 |
| 173 | 200x200 | 3146 | 3146 | 3146 | 0 | 79 | 21 | -58 | 2.233 | 2.724 | 0.491 |
| 174 | 200x200 | 367593 | 367593 | 367593 | 0 | 1140 | 686 | -454 | 18.178 | 49.652 | 31.474 |
| 175 | 200x200 | 445759 | 445759 | 445759 | 0 | 1183 | 857 | -326 | 14.689 | 48.048 | 33.359 |
| 176 | 200x200 | 371726 | 371726 | 371726 | 0 | 877 | 719 | -158 | 12.34 | 42.094 | 29.754 |
| 177 | 200x200 | 358538 | 358538 | 358538 | 0 | 1223 | 665 | -558 | 12.706 | 38.568 | 25.862 |
| 178 | 200x200 | 366224 | 366224 | 366224 | 0 | 1158 | 688 | -470 | 13.183 | 39.291 | 26.108 |
| 179 | 200x200 | 397794 | 397794 | 397794 | 0 | 1317 | 843 | -474 | 30.904 | 47.344 | 16.44 |
| 180 | 200x200 | 376528 | 376528 | 376528 | 0 | 967 | 708 | -259 | 13.191 | 40.043 | 26.852 |
| 181 | 200x200 | 364957 | 364957 | 364957 | 0 | 1115 | 630 | -485 | 13.635 | 36.486 | 22.851 |
| 182 | 200x200 | 338649 | 338649 | 338649 | 0 | 1078 | 516 | -562 | 10.771 | 29.989 | 19.218 |
| 183 | 200x200 | 366604 | 366604 | 366604 | 0 | 1365 | 758 | -607 | 13.953 | 42.63 | 28.677 |
| 184 | 200x200 | 339813 | 339813 | 339813 | 0 | 1081 | 530 | -551 | 19.799 | 32.115 | 12.316 |
| 185 | 200x200 | 334207 | 334207 | 334207 | 0 | 1177 | 596 | -581 | 11.415 | 33.491 | 22.076 |
| 186 | 200x200 | 342507 | 342507 | 342507 | 0 | 1197 | 567 | -630 | 11.985 | 33.138 | 21.153 |
| 187 | 300x300 | 475548 | 475548 | 475548 | 0 | 1582 | 1126 | -456 | 40.331 | 118.11 | 77.779 |
| 188 | 300x300 | 510129 | 510129 | 510129 | 0 | 1752 | 1143 | -609 | 43.062 | 121.796 | 78.734 |
| 189 | 300x300 | 470294 | 470294 | 470294 | 0 | 1937 | 1157 | -780 | 37.783 | 128.977 | 91.194 |
| 190 | 300x300 | 504693 | 504693 | 504693 | 0 | 1717 | 1245 | -472 | 42.28 | 135.749 | 93.469 |
| 191 | 300x300 | 493322 | 493322 | 493322 | 0 | 2111 | 1185 | -926 | 43.463 | 124.978 | 81.515 |
| 192 | 300x300 | 482604 | 482604 | 482604 | 0 | 1877 | 1200 | -677 | 41.913 | 127.808 | 85.895 |
| 193 | 300x300 | 474210 | 474210 | 474210 | 0 | 2038 | 1019 | -1019 | 44.383 | 108.199 | 63.816 |
| 194 | 300x300 | 522223 | 522223 | 522223 | 0 | 2284 | 1386 | -898 | 42.342 | 146.132 | 103.79 |
| 195 | 300x300 | 472639 | 472639 | 472639 | 0 | 1872 | 1079 | -793 | 39.156 | 115.055 | 75.899 |
| 196 | 300x300 | 486426 | 486426 | 486426 | 0 | 1897 | 1367 | -530 | 44.815 | 142.721 | 97.906 |
| 197 | 300x300 | 517029 | 517029 | 517029 | 0 | 2173 | 1223 | -950 | 42.236 | 128.89 | 86.654 |
| 198 | 300x300 | 499166 | 499166 | 499166 | 0 | 2029 | 1114 | -915 | 38.805 | 119.125 | 80.32 |
| 199 | 300x300 | 539414 | 539414 | 539414 | 0 | 2342 | 1511 | -831 | 42.039 | 158.702 | 116.66 |
| 200 | 300x300 | 487730 | 487730 | 487730 | 0 | 1954 | 1191 | -763 | 42.054 | 125.697 | 83.643 |