



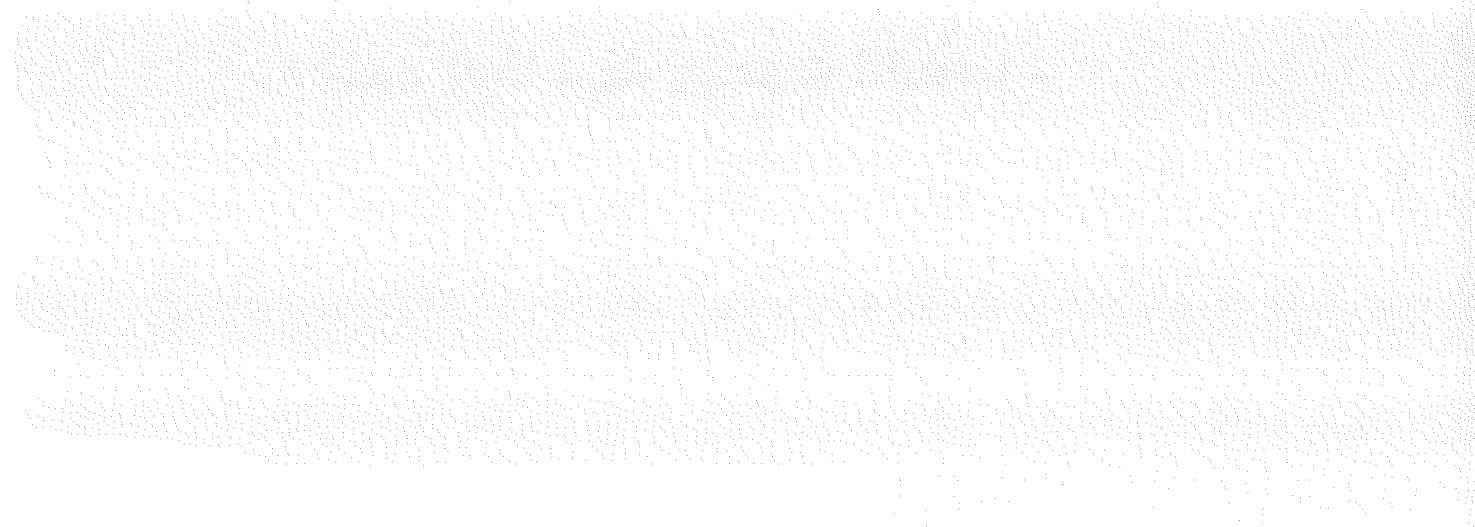
جامعة القاهرة
معهد الدراسات والبحوث الإحصائية



المؤتمر السنوي الثامن والأربعون للإحصاء
وعلوم الحاسب وبحوث العمليات

مجلد
بحوث عمليات

٢٣-٢٦ ديسمبر ٢٠١٣



- [29] Ross, Phillip J. "Taguchi Techniques for Quality Engineering (2nd ed'95)." (1995).
- [30] Deb, Kalyanmoy. "Multi-objective optimization." Multi-objective optimization using evolutionary algorithms (2001): 13-46.
- [31] Miettinen, Kaisa. Nonlinear Multiobjective Optimization. Vol. 12. Springer, 1999.
- [32] Miettinen, Kaisa. On The Methodology Of Multiobjective Optimization With Applicatins.1994
- [33] Steuer, Ralph E. Multiple Criteria Optimization: Theory, Computation, and Application. Malabar: Krieger, 1989.
- [34] Vira, Chankong, and Yacov Y. Haimes. Multiobjective Decision Making: Theory and Methodology. No. 8. North-Holland, 1983.
- [35] Zeleny, Milan. Multiple Criteria Decision Making. Edited by James L. Cochrane. Vol. 25. New York: McGraw-Hill, 1982.

- [19] Kim, Bosun, EsmaSenturk Gel, John W. Fowler, W. Matthew Carlyle, and JyrkiWallenius. "Evaluation of nondominated solution sets for k objective optimization problems: An exact method and approximations." *European journal of operational research* 173, no. 2 (2006): 565-582.
- [20] Sóbester, András, Stephen J. Leary, and Andy J. Keane. "On the design of optimization strategies based on global response surface approximation models." *Journal of Global Optimization* 33, no. 1 (2005): 31-59.
- [21] Schütze, Oliver, Alessandro Dell'Aere, and Michael Dellnitz. "On continuation methods for the numerical treatment of multi-objective optimization problems." In *Proceedings {schtze_et_al: DSP (2005): 349.*
- [22] Marler, R. Timothy, and Jasbir S. Arora. "Function-transformation methods for multi-objective optimization." *Engineering Optimization* 37, no. 6 (2005): 551-570.
- [23] Marler, R. Timothy, and Jasbir S. Arora. "Survey of multi-objective optimization methods for engineering." *Structural and multidisciplinary optimization* 26, no. 6 (2004): 369-395
- [24] Lepš, Matej. "Single and multi-objective Optimization in Civil Engineering with Applications." PhD diss., Czech Technical University, 2004.
- [25] Li, Yusheng, Georges M. Fadel, Margaret Wiecek, and Vincent Y. Blouin. "Minimum effort approximation of the Pareto space of convex bi-criteria problems." *Optimization and Engineering* 4, no. 3 (2003): 231-261.
- [26] Myers, Raymond H., and Christine M. Anderson-Cook. *Response surface methodology: process and product optimization using designed experiments*. Vol. 705. Wiley. com, 2002.
- [27] Deb, Kalyanmoy. "Multi-objective optimization." *Multi-objective optimization using evolutionary algorithms* (2001): 13-46.
- [28] Jin, Ruichen, Wei Chen, and Timothy W. Simpson. "Comparative studies of metamodelling techniques under multiple modelling criteria." *Structural and Multidisciplinary Optimization* 23, no. 1 (2001): 1-13.

- [10] Marler, R. Timothy, and Jasbir S. Arora. "The weighted sum method for multi-objective optimization: new insights." *Structural and multidisciplinary optimization* 41, no. 6 (2010): 853-862.
- [11] Gadallah, Mohamed H. "Optimization Formulation Based on Limited Data and RSM: An Approximation." In *Proceedings of the World Congress on Engineering*, vol. 1. 2009.
- [12] Jang, Beom-Seon, Dae-EunKo, Yong-Suk Suh, and Young-Soon Yang. "Adaptive approximation in multi-objective optimization for full stochastic fatigue design problem." *Marine Structures* 22, no. 3 (2009): 610-632.
- [13] Grandoni, Fabrizio, R. Ravi, and Mohit Singh. "Iterative rounding for multi-objective optimization problems." In *Algorithms-ESA 2009*, pp. 95-106. Springer Berlin Heidelberg, 2009.
- [14] Branke, Jürgen, Kalyanmoy Deb, KaisaMiettinen, and Roman Slowinski, eds. *Multiobjective optimization: Interactive and evolutionary approaches*. Vol. 5252. Springer, 2008.
- [15] Liu, G. P., X. Han, and C. Jiang. "A novel multi-objective optimization method based on an approximation model management technique." *Computer Methods in Applied Mechanics and Engineering* 197, no. 33 (2008): 2719-2731.
- [16] Janga Reddy, M., and D. Nagesh Kumar. "An efficient multi-objective optimization algorithm based on swarm intelligence for engineering design." *Engineering Optimization* 39, no. 1 (2007): 49-68.
- [17] Goel, Tushar, RajkumarVaidyanathan, Raphael T. Haftka, Wei Shyy, Nestor V. Queipo, and Kevin Tucker. "Response surface approximation of Pareto optimal front in multi-objective optimization." *Computer Methods in Applied Mechanics and Engineering* 196, no. 4 (2007): 879-893.
- [18] Janga Reddy, M., and D. Nagesh Kumar. "An efficient multi-objective optimization algorithm based on swarm intelligence for engineering design." *Engineering Optimization* 39, no. 1 (2007): 49-68.

Reference

- [1] Shahraki, AmenehForouzandeh, and RassoulNoorossana. "A Combined Algorithm for Solving Reliability-based Robust Design Optimization Problems." *Journal of mathematics and computer Science* 7 (2013) 54 - 62
- [2] Chen, Guodong, Xu Han, Guiping Liu, Chao Jiang, and Ziheng Zhao. "An efficient multi-objective optimization method for black-box functions using sequential approximate technique." *Applied Soft Computing* 12, no. 1 (2012): 14-27.
- [3] Kaveh, A., and K. Laknejadi. "A novel hybrid charge system search and particle swarm optimization method for multi-objective optimization." *Expert Systems with Applications* 38, no. 12 (2011): 15475-15488.
- [4] Lu, Songfeng, and Chengfu Sun. "Quadratic approximation based differential evolution with valuable trade off approach for bi-objective short-term hydrothermal scheduling." *Expert Systems with Applications* 38, no. 11 (2011): 13950-13960.
- [5] Guillén-Gosálbez, Gonzalo. "A novel MILP-based objective reduction method for multi-objective optimization: Application to environmental problems." *Computers & Chemical Engineering* 35, no. 8 (2011): 1469-1477.
- [6] Lu, Songfeng, and Chengfu Sun. "Quadratic approximation based differential evolution with valuable trade off approach for bi-objective short-term hydrothermal scheduling." *Expert Systems with Applications* 38, no. 11 (2011): 13950-13960.
- [7] Wang, X. D., C. Hirsch, Sh Kang, and C. Lacor. "Multi-objective optimization of turbomachinery using improved NSGA-II and approximation model." *Computer Methods in Applied Mechanics and Engineering* 200, no. 9 (2011): 883-895.
- [8] Wagner, Tobias, Michael Emmerich, André Deutz, and Wolfgang Ponweiser. "On expected-improvement criteria for model-based multi-objective optimization." In *Parallel Problem Solving from Nature, PPSN XI*, pp. 718-727. Springer Berlin Heidelberg, 2010.
- [9] Yaman, Sibel, and Chin-Hui Lee. "A Comparison of Single-and Multi-Objective Programming Approaches to Problems with Multiple Design Objectives." *Journal of Signal Processing Systems* 61, no. 1 (2010): 39-50.

Table 2: Summary of test problems using RSM

Prob	Domain	F* Reference	F*RSM	Type of problem
01	$X=[0,3]$, $y=[0,3]$	$X=2.1$ $y=1.788$ $F^*=0.026$	$X=2.1$ $y=1.606$ $F^*=-22.19$	Minimization
02	Domain= $[-50,50]$	$X=-.25$ $y=-0.036$ $F^*=-50$	$X=-2.764$ $y=-0.114$ $F^*=1.76$	Minimization
03	$X=[-1,1]$, $y=[-4,4]$	$X=-.04$ $y=.04$ $F^*=-12.948$	$x=-.768$ $y=-.28$ $F^*=-12.997$	Minimization
04	$X=[0,10]$, $y=[0,4]$	$X=-1.01$ $y=1.01$ $F^*=-0$	$X=-1.01$ $y=1.01$ $F^*=-0$	Minimization
05	$X, y > 0$	$X=0$ $y=0$ $F^*=0$	$X=-0$ $y=-0$ $F^*=0$	Minimization
06	Domain= $[-30,30]$	$X=-1.515$ $y=-.303$ $F^*=-.194$	$X=-.303$ $y=-.303$ $F^*=-.014$	Minimization
07	Domain= $[-400,400]$	$X=2.1$ $y=2.01$ $F^*=-16.471$	$x=2.1$ $y=-2.01$ $F^*=-17.026$	Minimization
08	Domain= $[-2,2]$	$X=-2$ $y=2$ $F^*=14$	$X=-2$ $y=2$ $F^*=14$	Minimization
09	Domain= $[-2,2]$	$X=.85$ $y=.7$ $F^*=0$	$X=.85$ $y=.7$ $F^*=0$	Minimization
10	Domain= $[0,1]$	$X=.815$ $y=.906$ $z=0.127$ $F^*=2.68$	$X=.965$ $y=.1576$ $z=.97$ $F^*=5.23$	Maximization
11	$X=[0, 8]$, $y=[0, 2.5]$	$X=8$ $y=2.5$ $F^*=2.15$	$X=8$ $y=2.5$ $F^*=1.951$	Maximization
12	$X=[0,4]$, $y=[0,4]$	$X=0$ $y=4$ $F^*=4$	$X=0$ $y=4$ $F^*=4$	Maximization
13	$X=[0,1]$, $y=[0,1], z=[0,3]$	$X=0.092$ $y=.194$ $z=1.6$ $F^*=6.874$	$X=0.168$ $y=.458$ $z=1.83$ $F^*=7.932$	Maximization
14	$X=[0,1.5]$, $y=[0,2.5], z=[0,5]$	$X=.268$ $y=1.786$ $z=.85$ $F^*=3.44$	$X=2$ $y=.787$ $z=1$ $F^*=6.881$	Maximization
15	$X=[0,5]$, $y=[0,3]$	$X=5$ $y=2$ $F^*=12$	$X=5$ $y=2$ $F^*=12$	Maximization
16	$X=[0,1]$, $y=[0,1], z=[0,3]$	$X=.142$ $y=.844$ $z=1.83$ $F^*=8.028$	$X=.142$ $y=.844$ $z=1.8$ $F^*=8.028$	Maximization
17	$X=[0,800]$, $y=[0,600], z=[0,600]$	$X=467.2$ $y=32.2$ $z=387.5$ $F^*=822586$	$X=467.2$ $y=32.2$ $z=387.5$ $F^*=822586$	Maximization
18	$X, y, z=[0,2]$	$X=.885$ $y=.913$ $z=.796$ $F^*=4.276$	$X=.885$ $y=.913$ $z=.796$ $F^*=4.276$	Maximization
19	$X, y, z=[0,3]$	$X=2.602$ $y=.171$ $z=.205$ $F^*=2.602$	$X=2.602$ $y=.171$ $z=.205$ $F^*=2.602$	Maximization
20	$X=[0,10]$, $y=[0,10]$	$X=3.918$ $y=6$ $F^*=21.918$	$X=3.918$ $y=6$ $F^*=21.918$	Maximization

Continue			
Prob	Reference	Domain	Optimal solution by reference
10	Simon Huband (2011)	Domain=[0,1]	X=.815 y=.906 ,z=0.127 F*=2.68
11	Milan Zeleny (1982).	X=[0,8] , y=[0,2.5]	X=8 y=2.5 F*=2.15
12	Ralph E. Stuer	X=[0,4] , y=[0,4]	X=0 y=4 F*=4
13	Milan Zeleny (1982).	X=[0,1] , y=[0,1],z=[0,3]	X=0.092 y=.194 z=1.6 F*=6.874
14	Milan Zeleny (1982).	X=[0,1.5] , y=[0,2.5], z=[0,5]	X=.268 y=1.786 z=.85 F*=3.44
15	Milan Zeleny (1982).	X=[0,5] , y=[0,3]	X=5 y=2 F*=12
16	Milan Zeleny (1982).	X=[0,1] , y=[0,1],z=[0,3]	X=.142 y=.844 z=1.83 F*=8.028
17	Milan Zeleny (1982).	X=[0,800] , y=[0,600], z=[0,600]	X=467.2 y=32.2 z=387.5 F*=822586
18	Milan Zeleny (1982).	X, y, z=[0,2]	X=.885 y=.913 z=.796 F*=4.276
19	Ralph E. Stuer (1989)	X=[0,2] y, z=[0,3]	X=2.602 y=171 z=.205 F*=2.602
20	Milan Zeleny (1982).	X=[0,10] , y=[0,10]	X=3.918y= 6 F*=21.918

5. Conclusion

After we had solved 20 problems and got the results shown in Table II, which are considered to be good results for RSM in dealing with multi-objectives optimization , we can conclude that the use of approximation model dealing with such problems is an effective and successful methods and the results in table 2 prove that ;

RSM solutions are better than Original solution for problems 1, 6, 7,10, ,13 and

14, RSM solutions = Original solution for problems, 3,4,5,8,9,12,15,16,17,18,19 and 20 , Original solution are better than RSM solutions for problem number 2 and 11 only.

4- The Comparison to determine the priority

After we had solved last problem we get The minimum values of f^* is $f^*_{\min(\text{res})}$ for $(f_1, f_2) = -22.19$ in $w=0.0$ (global optimal solution)

This means that the f_2 is the first objective, because the weight of f_2 is 1.0 and weight of f_1 is 0.0 and get the minimum value of f^* , and the second value of f^* is $f^*_{\min(\text{res})}$ for $(f_2, f_3) = -22.19$ in $w=1.0$, This means that the f_2 is the first objective, because the weight of f_2 is 1.0 and weight of f_3 is 0.0 and get the minimum value of f^* and the third value of f^* is $f^*_{\min(\text{res})}$ for $(f_1, f_3) = 0.102$ in $w=1.0$, This means that the f_1 is the first objective, because the weight of f_1 is 1.0 and weight of f_3 is 0.0 and get the minimum value of f^* From compeering between the results of $f^*_{\min(\text{res})}$ for (f_1, f_2) , (f_1, f_3) and (f_2, f_3) in this problem f_2 take the first priority, f_1 take the second priority and f_3 is third priority.

We have 20 Multi-objective problems with three objectives as table 1.

Table 1: Summary of problems tested.

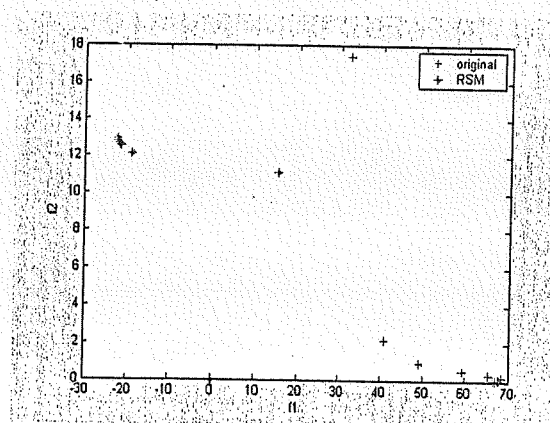
Prob	Reference	Domain	Optimal solution by reference
1	R.Timothy(2005)	$X=[0,3]$, $y=[0,3]$	$X=2.1$ $y=1.788$ $F^*=0.026$
2	Simon Huband (2011)	Domain= $[-50,50]$	$X=-.25$ $y=-0.036$ $F^*=-50$
3	(Carlos.A.C.andCartes,N.C. (2002)	$X=[-1,1]$, $y=[-4,4]$	$X=-.04$ $y=.04$ $F^*=-12.948$
4	(Vira Chankong and Yacov Y.Haimes) (1983)	$X=[0,10]$, $y=[0,4]$	$X=-1.01$ $y=1.01$ $F^*=-0$
5	(Vira Chankong and Yacov Y.Haimes (1983.)	$X, y > 0$	$X=0$ $y=0$ $F^*=0$
6	Simon Huband (2011)	Domain= $[-30,30]$	$X=-1.515$ $y=-.303$ $F^*=-.194$
7	Simon Huband (2011)	Domain= $[-400,400]$	$X=2.1$ $y=2.01$ $F^*=-16.471$
8	Simon Huband (2011)	Domain= $[-2,2]$	$X=-2$ $y=2$ $F^*=14$
9	Simon Huband (2011)	Domain= $[0,1]$	$X=.85$ $y=.7$ $F^*=0$

3- Similarly convert F_2, F_3 to single objective by

$$\text{Minimize } f(x) = W F_2 + (1 - W) F_3$$

Solve f_2, f_3

W	X	y	f^*	X(res)	y(res)	f^*_{res}
0.00	1.421	2.667	11.276	2.100	2.697	11.077
0.10	1.676	2.242	10.658	2.100	1.879	8.968
0.20	1.824	2.152	9.628	2.100	1.727	5.661
0.30	1.973	2.121	8.506	2.100	1.697	2.229
0.40	2.058	2.091	7.341	2.100	1.667	-1.239
0.50	2.100	2.061	6.152	2.100	1.636	-4.715
0.60	2.100	2.000	4.949	2.100	1.636	-8.210
0.70	2.100	1.970	3.736	2.100	1.636	-11.704
0.80	2.100	1.939	2.512	2.100	1.636	-15.199
0.90	2.100	1.879	1.276	2.100	1.636	-18.694
1.00	2.100	1.788	0.026	2.100	1.606	-22.190



Pareto frontier by RSM and original for F_2, F_3

W	X	y	f^*	X(res)	y(res)	f^*_{res}
1.00	2.100	1.788	0.026	2.100	1.606	-22.190

F^*_{min} for real case = 0.0224 $f^*_{min(res)}$ for $(f_1, f_2) = -22.19$ at $w = 0$

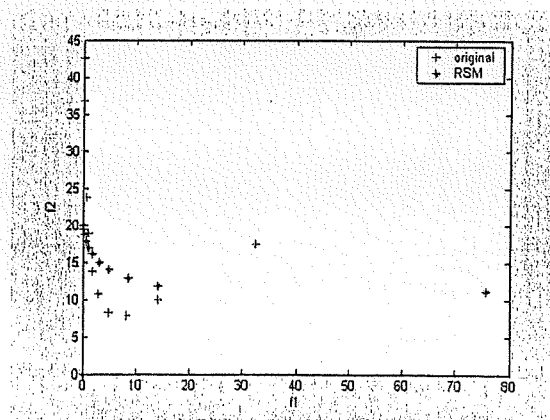
$f^*_{min(res)}$ for $(f_1, f_3) = 0.102$ at $w = 1$ $f^*_{min(res)}$ for $(f_2, f_3) = -22.19$ at $w = 1$

2- Similarly convert F_1, F_3 to single objective by

$$\text{Minimize } f(x) = WF_1 + (1 - W)F_3$$

Solve f_1, f_3

W	X	Y	f^*	X(res)	y(res)	f^*_{res}
0.00	1.421	2.667	11.276	2.100	2.697	11.077
0.10	1.039	2.303	12.163	1.039	2.303	12.067
0.20	0.891	2.030	11.989	0.912	2.030	11.940
0.30	0.806	1.788	11.278	0.806	1.788	11.255
0.40	0.721	1.636	10.213	0.742	1.606	10.204
0.50	0.679	1.485	8.892	0.679	1.455	8.892
0.60	0.636	1.333	7.381	0.636	1.333	7.384
0.70	0.594	1.242	5.720	0.594	1.242	5.724
0.80	0.552	1.152	3.940	0.552	1.152	3.944
0.90	0.530	1.061	2.061	0.530	1.061	2.062
1.00	0.509	1.000	0.102	0.509	1.000	0.102

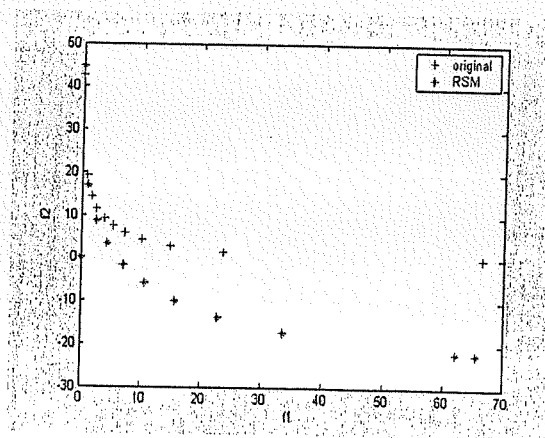


Pareto frontier by RSM and original for F_1, F_3

W	X	y	f^*	X(res)	y(res)	f^*_{res}
1.00	0.509	1.000	0.102	0.509	1.000	0.102

Solve F_1, F_2

W	X	Y	f^*	X(res)	y(res)	f^*res
0.00	2.100	1.788	0.026	2.100	1.606	-22.190
0.10	1.421	1.758	3.595	2.058	1.606	-13.427
0.20	1.209	1.758	5.234	1.633	1.606	-7.058
0.30	1.061	1.727	6.088	1.421	1.606	-2.774
0.40	0.955	1.697	6.429	1.252	1.606	0.259
0.50	0.870	1.667	6.371	1.103	1.576	2.309
0.60	0.785	1.636	5.967	0.976	1.545	3.544
0.70	0.700	1.606	5.247	0.848	1.545	4.011
0.80	0.636	1.515	4.186	0.742	1.485	3.730
0.90	0.573	1.394	2.681	0.615	1.394	2.632
1.00	0.509	1.000	0.102	0.509	1.000	0.102



Pareto frontier by RSM and original for F_1, F_2

W	X	y	f^*	X(res)	y(res)	f^*res
0.00	2.100	1.788	0.026	2.100	1.606	-22.190

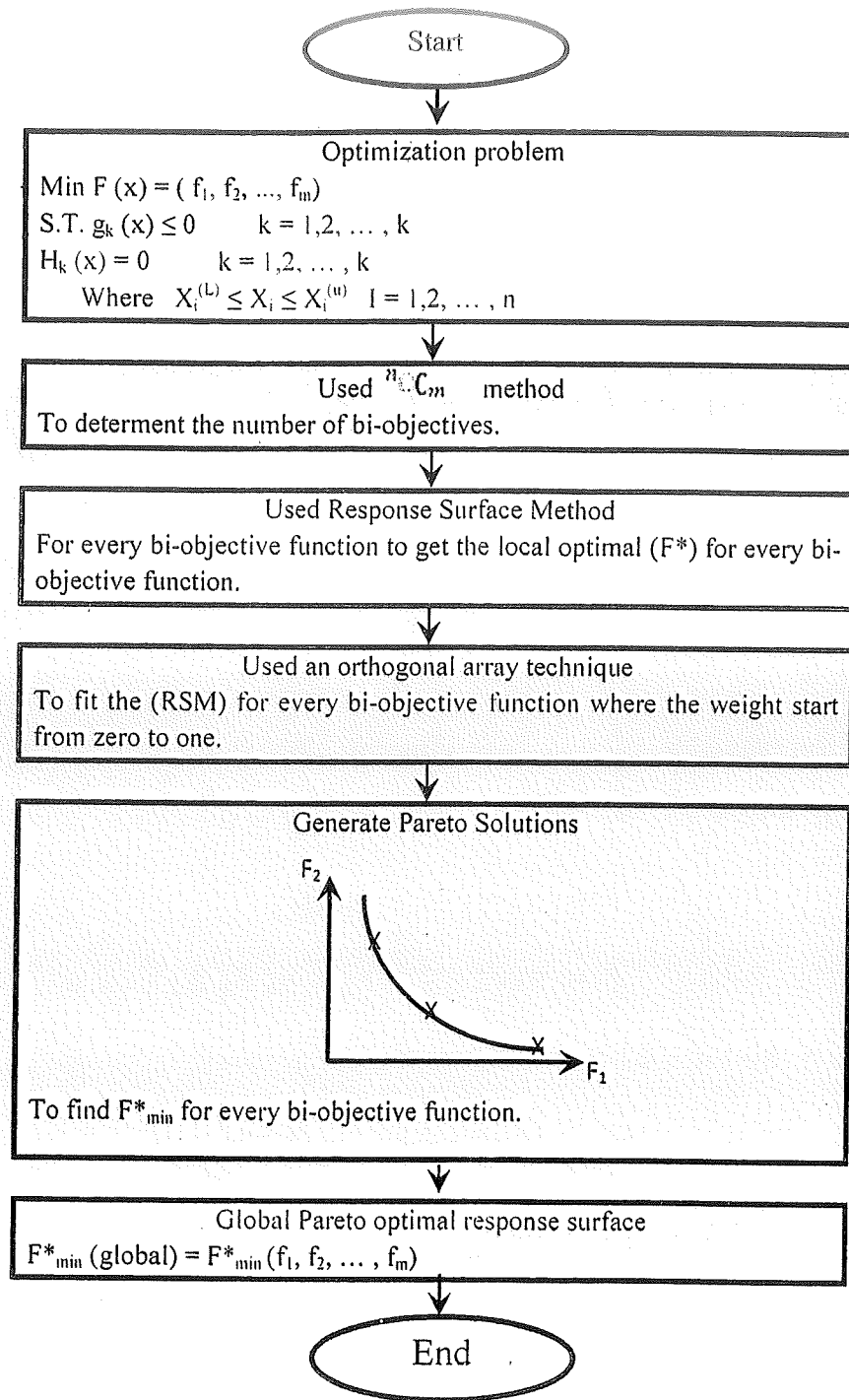


Figure 1. Pareto Optimal Front RSM approximation by an Optimization Formulation based on a set of bi – objectives.

The second order model is widely used in response surface methodology for several reasons .Among these reasons are the following:

1-the second –order model is very flexible. It can take on a wide variety of functional forms, so it will often work well as an approximation to the true response surface.

2-It is easy to estimate the parameters (the β 's) in the second –order model.

3- There is considerable practical experience indicating that the secondorder models work well in solving real response surface problems. (Raymond and Montgomery, 2002).

4. Generating the Pareto front response surface approximation

The methodology used here for generating the Pareto front response surface approximation can be illustrated with the help of Figure 1.

5-Numerical examples

R.Timothy (2005)

$$\text{Minimize:- } (x) \quad F_1 = 25 (X - 0.5)^2 + (2y - 2)^2 + 0.1$$

$$F_2 = [(X - 2.5)^2 + 4(y + 108)^2]^2$$

$$F_3 = -(X - 2)^4 + 1.5 (y - 2.8)^2 + 0.3 X y + 10$$

$$\text{Subject to :- } g_1 = (X - 2.1) - 0.08 (2.2 - y)^2 \leq 0 \quad g_2 = -x \leq 0 \quad g_3 = -y \leq 0$$

$$g_4 = y - 3.0 \quad X \leq 0 \quad X \geq 0, y \geq 0$$

Solution steps

1- convert F_1, F_2 to single objective by

$$\text{Minimize } f(x) = W F_1 + (1 - W) F_2$$

$$\text{Subject to: - } g_1 = (X - 2.1) - 0.08 (2.2 - y)^2 \leq 0 \quad g_2 = -x \leq 0 \quad g_3 = -y \leq 0$$

$$g_4 = y - 3.0 \quad X \leq 0 \quad X \geq 0, y \geq 0$$

3. Methodology:

Response Surface Methodology (RSM). Is not an algorithm, it is a set of statistical grouped together to analyze a problem. RSM is a collection of statistical and mathematical techniques useful for developing, improving and optimizing process. It also has important applications in the design. Development and formulation of new products as well as in the improvement of existing product designs.

Definition: Response Surface Methodology (RSM) is a collection of statistical and mathematical techniques useful for developing, improving, and optimizing processes. It also has important applications in the design, development, and formulation of new products, as well as in the improvement of existing product designs. (Raymond and Montgomery, 2002).

In many cases, either the first order or the second order model is used as follows:

$$y = \beta_0 + \sum_{i=1}^n \beta_i x_i \quad (3)$$

$$y = \beta_0 + \sum_{i=1}^n \beta_i x_i + \sum_{i=1}^n \beta_{ii} x_i^2 + \sum_{i < j} \sum_{j=1}^n \beta_{ij} x_i x_j \quad (4)$$

Parameter β_0 , β_i , β_{ii} , and β_{ji} of the polynomials in Eqns.(5) and (6) are determined through least square regression given by the following Eqn :

$$\beta = [X' X]^{-1} X' f^* \quad (5)$$

Where x is an $n \times p$ the design matrix of sample data points. X' is its transpose, and f^* is a column vector an $n \times 1$, containing the values of the response at each sample point. It is to see that $X' X$ is a $p \times p$ symmetric matrix and $X' y$ is a $p \times 1$ column vector.

An objective vector $z^* \in Z$ is Pareto optimal if there does not exist another objective vector $z \in Z$ such that $z_i \leq z_i^*$ for all $i = 1, \dots, k$ and $z_j < z_j^*$ for at least one index j ; or equivalently, z^* is Pareto optimal if the decision vector corresponding to it is Pareto optimal. (kaisa 1999)

Definition 2. A decision vector $x^* \in S$ is a locally Pareto optimal solution if there exists $y > 0$ such that x^* is Pareto optimal in $S \cap B(x^*, y)$.

Let the multiobjective optimization problem be convex. Then every locally Pareto optimal solution is also globally Pareto optimal. (kaisa 1999)

Definition 3: Search space or design space is the set of all possible combinations of the design parameters. If all design variables are real, the design space is given as $S \in R^N$, (N is the number of design variables). The feasible domain S is the region in design space where all constraints are satisfied. (Tushar Goel et al 2007)

Definition 4: The function space representation of all the solutions in the Pareto optimal set is the Pareto optimal front. When there are two objectives, the Pareto optimal front is a curve, when there are three objectives, the Pareto optimal front is represented by a surface and if there are more than three objectives, it is represented by a hyper-surface. (Tushar Goel et al 2007)

Weighting Method

In this way , the multiobjective functions are transformed into a single objective function . They suppose that the weighting coefficients w_i are real numbers such that $w_i \geq 0$ for all $i = 1, \dots, k$. It is also usually supposed that the weight are normalized , that is , $\sum_{i=1}^k w_i = 1$. Then the multiobjective optimization problem (1) is modified into the following problem . (kaisa 1999)

$$\text{Minimize } \sum_{i=1}^k w_i f_i(x)$$

$$\text{Subject to : } x \in X \quad (2)$$

Where $w_i \geq 0$ for all $i = 1, \dots, k$, and

$$\sum_{i=1}^k w_i = 1$$

As non-convex and disconnected Pareto-optimal fronts, are also involved. In multi-criterion decision making, no single solution can be termed as the optimum solution to the multiple conflicting objectives, as a MOOP is amenable to a number of trade-off optimal solutions (Deb 2001). For this purpose, multi-objective optimization generates a Pareto front, which is a set of non-dominated solutions for problems with more than one objective. The major goal of any MOOP algorithm is to generate a well-distributed true Pareto-optimal front or surface. A brief description of multi-objective optimization concepts is presented in the next section. The paper is organized as follows: Section 2 gives the Basic terminology used in the paper. Section 3 Methodology, Section 4 presents generating the Pareto front response surface approximation. Section 5 numerical example Finally, Section 6 summarizes the main conclusions of the paper.

2. Basic terminology

In this section, some terms used in this paper are defined

Problem definition (MOOP)

$$\min(f_1(x), f_2(x), \dots, f_k(x)) \quad (1)$$

$$\text{Subject to : } x \in S,$$

where the integer $k \geq 2$ is the number of objectives functions $f_i: R^n \rightarrow R$ we denote the vector of objective functions by $f(x) = (f_1(x), f_2(x), \dots, f_k(x))^T$. The decision variable vectors $x = (x_1, x_2, \dots, x_n)^T$ belong to the (nonempty) feasible region (set) S , which is a subset of the decision variable space R^n . (Kaisa 1994)

Pareto Optimality

Definition 1 . A decision vector $x^* \in S$ is Pareto optimal if there does not exist another decision vector $x \in S$ such that $f_i(x) \leq f_i(x^*)$ for all $i = 1, \dots, k$ and $f_j(x) < f_j(x^*)$ for at least one index j .

Treating Multi-Objective Problems Using Special Approximations

¹ S. H. Eldeeb, ² M. H. Gadallah, ³ O. S. Soliman

And ⁴ H. A. Khalifa

Abstract

Multi-objective optimization is a powerful technique to show the Pareto Frontier for systems with many objectives. Interpretation of the Pareto Front is possible for 3 dimension problems, but the multi-objectives problem with more than two objectives solved by the priority of the objectives from the owner or the weight for every objectives to determent the Pareto frontier and optimal solution. This solution is not complete because the solution depends on the priority of objectives , Accordingly, In this paper we use one of the important methods that deal with multi-objectives which "Response Surface Methodology" (RSM) with weighted method to transfer multi-objectives problem into a series of bi-objective optimization problems to determine the Pareto frontier and optimal solution, and So that we can in the end, have many of the Pareto frontier and optimal solution depending on the number of sub problems and not depending on the priority of objectives to satisfy the following purposes:

The validity of the method to approximate the original multi-objective problem into a series of bi-objective problems, Possibility of seeing all the Pareto optimal solution by dividing the problem into a series of bi-objective, The computational savings related to this transformation, especially with large size problems, The Pareto frontier resulting from the bi-objective formulation versus the original multi-objective formulation.

. **Key words:** Multi-objectives optimization, Pareto optimal. Response Surface Methodology.

1. Introduction

Multi-objective optimization techniques play an important role in engineering design, resources optimizations, and many other fields. Their main purpose is to find a set of best solutions from which a designer or a decision-maker can choose a solution to derive maximum benefit from the available resources. Various objectives of a multi-objective optimization problem (MOOP) often conflict and/or compete with one another. Complex relationships in the decision space and/or function space, such

1,4- Department of Operations Research, Institute of Statistical Studies & Research, Cairo University , Cairo, Egypt

2- Faculty of Engineering, Cairo University, 3-- Faculty of computers & information , Cairo University Cairo, Egypt

- [12] Prakash S., "UNIT-3: Transportation Problem", Visvesvaraya Technological University, e-Learning Centre, Mysore, 06IP/IM74 Operations Research, 2013, pp. 1-39.
- [13] Fogiel M., The Operations Research Problem Solver, Research and Education Association, New York, N. Y. 10018, 1987.

7. References

- [1] Pandian P. and Natarajan G., "A New Algorithm for Finding a Fuzzy Optimal Solution for Fuzzy Transportation Problems", Applied Mathematical Sciences, Vol. 4, Issue No. 2, 2010, pp. 79 – 90.
- [2] Samuel A., "Improved Zero Point Method (IZPM) for the Transportation Problems", Applied Mathematical Sciences, Vol. 6, Issue No. 109, 2012, pp. 5421 – 5426.
- [3] Taha H., Operations Research an Introduction, Macmillan Publishing Co., Inc., 1982.
- [4] Sharma G., Abbas S. and Gupta V., "Optimum Solution of Transportation Problem with the Help of Zero Point Method", International Journal of Engineering Research & Technology, Vol. 1, Issue 5, 2012, pp. 1-6.
- [5] Venkatasubbaiah K, Acharyulu S and Chandra Mouli K, "Fuzzy Goal Programming Method for Solving Multi-Objective Transportation Problems ", Global Journal of Research in Engineering, Vol. 11, Issue No. 3, 2011, pp. 4-10.
- [6] Samuel A. and Venkatachalapathy M., "Modified Vogel's Approximation Method for Fuzzy Transportation Problems", Vol. 5, Issue No. 28, 2011, pp. 1367 - 1372.
- [7] Hillier F. and Lieberman G., Introduction to Operations Research, McGraw-Hill companies, Inc., 2001.
- [8] Quddoos A., Javaid S. and Khalid M., "A New Method for Finding an Optimal Solution for Transportation Problems", International Journal on Computer Science and Engineering, Vol. No. 4, Issue No. 7, 2012, pp. 1271-1274.
- [9] Pandian P. and Natarajan G., "A New Method for Finding an Optimal Solution of Fully Interval Integer Transportation Problems", Applied Mathematical Sciences, Vol. 4, Issu No. 37, 2010, PP. 1819-1830.
- [10] Adlakha V. and kowalski K., "A Quick Sufficient Solution to the More- for-Less Paradox in the Transportation Problem", Omega International Journal of Management Science, Vol. 26, Issue No. 4, 1998, pp. 541-547.
- [11] Reeb J. and Leavengood S., "Transportation Problem: A Special Case for Linear Programming Problems", Operations Research EM 8779, 2002, pp. 1-36.

Table 30

	d_1	d_2	d_3	d_4	d_5	
S_1	11 (6) 200 0	13 (7) 50 0	17 9	14 9	0 5	250
S_2	16 0	18 (8) 125 0	14 1	10 (5) 175 0	0 (3) 50 0	350
S_3	21 5	24 6	13 (1) 275 0	10 (4) 125 0	0 0	400
S_4	15 0	17 (9) 50 0	21 9	18 9	0 1	50
S_5	21 5	23 5	19 6	15 5	0 (2) 75 0	75
Demand	200	225	275	300	125	1125

The optimal allocation of such example is: $x_{11} = 200$, $x_{12} = 50$, $x_{22} = 125$, $x_{24} = 175$, $x_{25} = 50$, $x_{33} = 275$, $x_{34} = 125$, $x_{42} = 50$ and $x_{55} = 75$ with the total cost = 12525.

5. Comparison between the proposed algorithm and the ZPM and the IZPM

We have chosen 40 transportation problems from different references as follows: one problem from Taha [2], one problem from Hillier and Lieberman [7], one problem from Qudoods et al. [8], four problems from Pandian and Natarajan [9], two problems from Sharma et al. [4], two problems from Adlakha and Kowalski [10], four problems from Reeb and Leavengood [11], twelve problems from Prakash [12], six problems from Fogiel [13] and eight problems from other references.

We solved these 40 problems using the proposed algorithm and compared the solutions with the IZPM method. The conclusion was that using our algorithm gives the optimal solution for the 40 problems, while solving the same problems using the ZPM and the IZPM gives optimal solution for 34 problems only.

6. Conclusion

In this paper an algorithm for solving transportation problems has been proposed to handle the drawbacks in the ZPM and the IZPM. This algorithm presented a modified version of both the ZPM and the IZPM. The algorithm overcomes the arbitrary selection criterion in ZPM and IZPM by proposing a new criterion for the selection of the zero cells depending on the original transportation cost matrix. This algorithm also adds a new condition to adjust the optimality criteria in both methods. The proposed algorithm gets an optimal solution for the transportation problems compared with the ZPM and the IZPM.

Table 28

	d_1	d_2	d_3	d_4	d_5	
S_1	11 (6) 200 0	13 (7) 50 0	17 9	14 9	0 5	250
S_2	16 0	18 (9) 125 0	14 1	10 (5) 225 0	0 0	350
S_3	21 5	24 6	13 (1) 275 0	10 (4) 75 0	0 (3) 50 0	400
S_4	15 0	17 (8) 50 0	21 9	18 9	0 1	50
S_5	21 5	23 5	19 6	15 5	0 (2) 75 0	75
Demand	200	225	275	300	125	1125

The optimal allocation of such example is: $x_{11} = 200$, $x_{12} = 50$, $x_{22} = 125$, $x_{24} = 225$, $x_{33} = 275$, $x_{34} = 75$, $x_{35} = 50$, $x_{42} = 50$ and $x_{55} = 75$ with the total cost = 12525.

In step 8 (c) the cells (2, 5) and (3, 5) have the same transportation costs. If we choose cell (2, 5) for allocation as presented in Table 29, instead of cell (3, 5) in Table 18, we get an alternative solution in Table 30.

Table 29

	d_1	d_2	d_3	d_4	d_5	
S_1	11 0	13 0	17 9	14 9	0 5	250
S_2	16 0	18 0	14 1	10 0	0 (3) 50 0	300
S_3	21 5	24 6	13 (1) 275 0	10 0	0 0	125
S_4	15 0	17 0	21 9	18 9	0 1	50
S_5	21 5	23 5	19 6	15 5	0 (2) 75 0	0
Demand	200	225	0	300	0	1125

Table 26

	d_1	d_2	d_3	d_4	d_5	
S_1	11 0	13 0	17 9	14 9	0 5	250
S_2	16 0	18 0	14 1	10 0	0 0	350
S_3	21 5	24 6	13 25	10 0	0 5	75
S_4	15 0	17 0	21 9	18 9	0 1	50
S_5	21 5	23 5	19 6	15 5	0 75	0
Demand	200	225	0	300	0	1125

Step 9: Since there is a single zero cell (3, 4) in row s_4 , then allocate it as presented in Table 27.

Table 27

	d_1	d_2	d_3	d_4	d_5	
S_1	11 0	13 0	17 9	14 9	0 5	250
S_2	16 0	18 0	14 1	10 0	0 0	350
S_3	21 5	24 6	13 25	10 0	0 5	0
S_4	15 0	17 0	21 9	18 9	0 1	50
S_5	21 5	23 5	19 6	15 5	0 75	0
Demand	200	225	0	225	0	1125

Repeating step 10: Since all supplies and demands are not satisfied, we still have Table 27.

Repeating steps from 7 to 10 until all supplies and demands are satisfied.

Step 11: Table 28 giving the optimal transportation plan:

Table 25

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	11 0	13 0	17 0	14 0	0 5	250
s_2	16 0	18 0	14 1	10 0	0 0	350
s_3	21 5	24 6	13 2.5	10 0	0 0	125
s_4	15 0	17 0	21 0	18 0	0 1	50
s_5	21 5	23 5	19 6	15 5	0 7.5	0
Demand	200	225	0	300	50	1125

Step 10: Since all supplies and demands are not satisfied, we still have Table 25.

Repeating Step 7: Selecting the largest cost (9) in Table 25, since there is more than one cell having the same cost, we select cell (1, 4).

Repeating step 8: Since the selected cell in step 7 is the cell (1,4), then there are two zeros in both the row s_1 and the column d_4 . Select from the reduced matrix the next largest cost (9) to the cost of that cell (4, 4) , and then there are two zeros in both the row s_4 and the column d_4 . Select from the reduced matrix the next largest cost (6) to the cost of that cell (3, 2), and then there are two zeros in both the row s_3 and the column d_2 . Select from the reduced matrix the next largest cost (5) to the cost of that cell (1, 5), and then there are two zeros in both the row s_1 and the column d_1 . Select from the reduced matrix the next largest cost (5) to the cost of that cell (3, 1)), and then there are two zeros in both the row s_3 and the column d_1 . Select from the reduced matrix the next largest cost (1) to the cost of that cell (4, 5)), and then there are two zeros in both the row s_4 and the column d_5 . We still have Table 25.

Step 8 (c): The next largest reduced cost is zero. Choosing the unallocated cell (3, 5) with the smallest transportation cost (0) in the original cost matrix gives Table 26.

Table 23

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	11 0	13 0	17 0	14 0	0 5	250
s_2	16 0	18 0	14 1	10 0	0 0	350
s_3	21 5	24 6	13 275	10 0	0 0	125
s_4	15 0	17 0	21 0	18 0	0 1	50
s_5	21 5	23 5	19 6	15 5	0 0	0
Demand	200	225	0	300	50	1125

Step 8: Since the selected cell in step 7 is the cell (1, 3), then there is one zero in the column d_3 , then we allocate the cell (3, 3) as presented in Table 24.

Table 24

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	11 0	13 0	17 0	14 0	0 5	250
s_2	16 0	18 0	14 1	10 0	0 0	350
s_3	21 5	24 6	13 275	10 0	0 0	125
s_4	15 0	17 0	21 0	18 0	0 1	50
s_5	21 5	23 5	19 6	15 5	0 0	0
Demand	200	225	0	300	50	1125

Step 9: Since there is a single zero cell (5, 5) in row s_5 , then allocate it as presented in Table 25.

Table 21

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	8	8	4	250
s_2	1	①	1	0	0	350
s_3	6	7	0	0	0	400
s_4	0	0	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Repeating Step 6: Subtracting the smallest cost (1) in the uncovered entries in Table 21, and adding it to the intersection of every two lines, we get Table 22.

Table 22

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	9	9	5	250
s_2	0	0	1	0	0	350
s_3	5	6	0	0	0	400
s_4	0	0	9	9	1	50
s_5	5	5	6	5	0	75
Demand	200	225	275	300	125	

Repeating step 4: Checking the first row, the zeros in that row of destinations d_1 , d_2 and d_5 with total demands = $200 + 225 + 125 = 550$ is greater than the supply s_1 (250). Also, if the same check is repeated for all the rows and all the columns the allocation condition will be satisfied. So, go to step 7.

Step 7: Selecting the largest cost (9) in Table 22, since there is more than one cell having the same cost, we select cell (1, 3) as presented in Table 23.

Step 6: Subtracting the smallest cost (4) in the uncovered entries in Table 18, and adding it to the intersection of every two lines, we get Table 19.

Table 19

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	8	8	4	250
s_2	1	1	1	0	0	350
s_3	6	7	0	0	0	400
s_4	0	0	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Repeating step 4: Checking the first row, we notice that the zeros in that row of destinations d_1 , d_2 and d_5 with total demands = $200 + 225 + 125 = 550$ is greater than the supply s_1 (250). Also, if the same check is repeated for all the rows and all the columns, we conclude that d_1 and d_2 in Table 19 are not satisfying the condition as illustrated in Table 20.

Table 20

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	8	8	4	250
s_2	1	1	1	0	0	350
s_3	6	7	0	0	0	400
s_4	0	0	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Repeating Step 5: Covering all zeros in Table 20 with the minimum number of horizontal and vertical lines and keeping the unsatisfied columns (d_1 and d_2) uncovered gives Table 21.

Table 16

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	4	4	0	250
s_2	5	5	1	0	0	350
s_3	10	11	0	0	0	400
s_4	4	4	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Step 4: Checking the first row, we notice that the zeros in that row of destinations d_1 , d_2 and d_5 with total demands = $200 + 225 + 125 = 550$ which is greater than the supply s_1 (250). Doing the same check to the rest of the rows and the columns, we conclude that d_1 and d_2 in Table 16 are not satisfying the condition as illustrated in Table 17.

Table 17

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	4	4	0	250
s_2	5	5	1	0	0	350
s_3	10	11	0	0	0	400
s_4	4	4	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Step 5: Covering all zeros in Table 17 with the minimum number of horizontal and vertical lines and keeping the unsatisfied columns (d_1 and d_2) uncovered gives Table 18.

Table 18

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	4	4	0	250
s_2	5	5	1	0	0	350
s_3	10	11	0	0	0	400
s_4	4	4	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Table 14

	d1	d2	d3	d4	Supply
s_1	2 (4) 200 0	2 (1) 800 0	2 0	4 3	1000
s_2	4 (5) 700 0	6 2	4 0	3 0	700
s_3	3 2	2 1	1 (3) 500 0	0 (2) 400 0	900
Demand	900	800	500	400	2600

The optimal allocation of such example is: $x_{11} = 200$, $x_{12} = 800$, $x_{21} = 700$, $x_{33} = 500$ and $x_{34} = 400$ with total cost = 5300.

4.2. Example 2

Using the proposed algorithm to solve the problem in Table 3 in section 2 gives the following:

Step 1: Since the problem is unbalanced, adding the dummy destination d_5 with demand = 125, we get Table 15.

Table 15

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	11	13	17	14	0	250
s_2	16	18	14	10	0	350
s_3	21	24	13	10	0	400
s_4	15	17	21	18	0	50
s_5	21	23	19	15	0	75
Demand	200	225	275	300	125	

Step 2: Since the minimum cost in every row is zero, we still have Table 15.

Step 3: Subtracting the minimum cost of each column from every entry in that column, we get Table 16.

Table 11, since there are more than one cell having the same costs, we select cell (2, 2), and allocate it as presented in Table 12.

Table 12

	d1	d2	d3	d4	Supply
s_1	2 0	2 800 0	2 0	4 3	200
s_2	4 0	6 2	4 0	3 0	700
s_3	3 2	2 1	1 0	0 0	900
Demand	900	0	500	400	2600

Step 9: Since there is not a single zero cell in all the rows and the columns, then go to step 7.

Repeating Step 7: The next largest reduced cost is zero in Table 11.

Repeating Step 8 (c): Choosing the unallocated cell (3, 4) with the smallest transportation cost (0) in the original transportation cost matrix, and allocate it as presented in Table 13.

Table 13

	d1	d2	d3	d4	Supply
s_1	2 0	2 800 0	2 0	4 3	200
s_2	4 0	6 2	4 0	3 0	700
s_3	3 2	2 1	1 0	0 400 0	500
Demand	900	0	500	0	2600

Step 9: Since all supplies and demands are not satisfied, we still have Table 13, repeat steps from 7 to 9 until all supplies and demands are satisfied.

Step 10: Table 14 giving the optimal transportation plan.

Table 3

	d_1	d_2	d_3	d_4	Supply
s_1	0	0	0	2	1000
s_2	1	3	1	0	
s_3	3	2	1	0	
Demand	900	800	500	400	2600

Step 5: Covering all zeros in Table 3 with the minimum number of horizontal and vertical lines and keeping the unsatisfied rows (s_2 and s_3) and columns (d_1 , d_2 and d_3) uncovered gives Table 4.

Table 4

	d_1	d_2	d_3	d_4	Supply
s_1	0	0	0	2	1000
s_2	①	3	1	0	700
s_3	3	2	1	0	900
Demand	900	800	500	400	2600

Step 6: Subtracting the smallest cost (1) in the uncovered entries in Table 4, and adding it to the intersection of every two lines, we get Table 5.

Table 5

	d_1	d_2	d_3	d_4	Supply
s_1	0	0	0	3	1000
s_2	0	2	0	0	700
s_3	2	1	0	0	900
Demand	900	800	500	400	2600

Repeating step 4: Checking the first row, we notice that the zeros in that row of destinations d_1 , d_2 and d_3 with total demands = $900 + 800 + 500 = 2200$ is greater than the supply $s_1 = 1000$. Also, if the same check is repeated for all rows and all columns the allocation condition will be satisfied. So, go to step 7.

Step 7: Selecting the largest cost (3) in Table 11.

Step 8: Since the selected cell in step 7 is the cell (1, 4), then there are two zeros in both the row s_1 and the column d_4 . Select from the reduced cost transportation matrix the next largest cost (2) in

cost in the original transportation cost matrix and go to step 9.

Step 9. If there is a single zero cell in any column or / and row, then allocate it.

Step 10. If all supplies and demands are not satisfied, go to step 7, otherwise go to step 11.

Step 11. The last table gives the optimal solution.

4. Illustrative examples

4.1.Example 1

Using the proposed algorithm to solve the problem in Table 1 in section 2 gives the following:

The results of applying the proposed algorithm can be described as follows:

Step 1: Since the problem is balanced, we still have Table 1.

Step 2: Subtracting the minimum cost of each row from every entry in that row, we get Table 2.

Table 2

	d_1	d_2	d_3	d_4	Supply
s_1	0	0	0	2	1000
s_2	1	3	1	0	700
s_3	3	2	1	0	900
Demand	900	800	500	400	2600

Step 3: Since the minimum cost in every column is zero, we still have Table 2.

Step 4: Checking the first row, we notice that the zeros in that row of destinations d_1 , d_2 and d_3 with total demands = $900 + 800 + 500 = 2200$ is greater than the supply $s_1 = 1000$. Therefore, the supply s_1 cannot satisfy the demands these three demands. Doing the same check to the rest of the rows and the columns, we conclude that d_1 , d_2 , d_3 , s_2 and s_3 are not satisfying the allocation condition as illustrated in Table 3.

there are another zero cells in the same row supplies (column demands). If we replace the arbitrary choice in the IZPM by choosing the smallest cost in the original transportation cost matrix which corresponds to one of the zero cells in the reduced transportation matrix, we get an optimal solution. The steps of the proposed algorithm are illustrated as follows:

- Step 1:** If the TP is unbalanced change it into a balanced one by adding a dummy row or column.
- Step 2:** Subtract the minimum cost of each row from all the entries of that row.
- Step 3:** Subtract the minimum cost of each column resulting from step 2 from all the entries of that column.
- Step 4:** If each row supply (column demand) is less than or equal to the sum of the demands (supplies) whose reduced costs are zeros in that row (column) and in the same time the sum of the row supplies (column demand) in these rows (columns) is less than or equal to the demand of that column (the supply of that row), then be sure there are another zero cells in the same row supplies (column demands), then go to step 7, otherwise, go to step 5.
- Step 5:** Cover all zeros of the reduced cost transportation table resulting from step 3 with the minimum number of horizontal and vertical lines and keep the unsatisfied rows or/and columns from step 4 uncovered.
- Step 6:** Subtract the smallest cost in the uncovered entries from each uncovered entries of the reduced cost transportation matrix, add it to the intersection of every two lines and go to step 4.
- Step 7:** Select the largest cost from the reduced cost transportation table . If that cost exists in more than one cell, then select any of these cells.
- Step 8:** If the selected cell in step 7 is the cell (i, j) then:
 - a) If there is one zero in the row i and / or the column j , then allocate the cell (cells) corresponding to this zero (zeros), go to step 9.
 - b) If there is more than one zero in the row i and column j then select from the reduced cost transportation matrix the next largest cost to the cost of the that cell (i, j) . Go to step 8 (a), otherwise go to step 8 (c).
 - c) If the next largest reduced cost is zero, then choose the unallocated cell with the smallest transportation

Table 5

	d_1	d_2	d_3	d_4	d_5	
S_1	11 (5) 75 0	13 (4) 175 0	17 9	14 9	0 5	250
S_2	16 (6) 125 0	18 0	14 1	10 (8) 175 0	0 (9) 50 0	350
S_3	21 5	24 6	13 (1) 275 0	10 0	0 (7) 125 0	400
S_4	15 0	17 (3) 50 0	21 9	18 9	0 1	50
S_5	21 5	23 5	19 6	15 5	0 (2) 75 0	75
Demand	200	225	275	300	125	1125

Table 6

	d_1	d_2	d_3	d_4	d_5	
S_1	11 0	13 0	17 9	14 9	0 5	250
S_2	16 0	18 (3) 225 0	14 1	10 0	0 0	125
S_3	21 5	24 6	13 (1) 275 0	10 0	0 0	125
S_4	15 0	17 3 0	21 9	18 9	0 1	0
S_5	21 5	23 5	19 6	15 5	0 (2) 75 0	0
Demand	200	0	0	300	50	1125

Table 7

	d_1	d_2	d_3	d_4	d_5	
S_1	11 (4) 200 0	13 0	17 9	14 X 9	0 5	50
S_2	16 0	18 (3) 225 0	14 1	10 (6) 75 0	0 (5) 50 0	0
S_3	21 5	24 6	13 (1) 275 0	10 (7) 125 0	0 0	0
S_4	15 0	17 3 0	21 9	18 X 9	0 1	50
S_5	21 5	23 5	19 6	15 5	0 (2) 75 0	0
Demand	0	0	0	100	0	1125

3. The proposed Algorithm

If we add a new optimality condition in the IZPM by also check if the sum of the row supplies (column demand) in these rows (columns) is less than or equal to the demand of that column (the supply of that row), then be sure

In order to illustrate this consider the unbalanced transportation problem, where s_i represents the supply of the source i and d_j represents the demand of the destination j as presented in Table 3.

Table 3

	d_1	d_2	d_3	d_4	Supply
s_1	11	13	17	14	250
s_2	16	18	14	10	350
s_3	21	24	13	10	400
s_4	15	17	21	18	50
s_5	21	23	19	15	75
Demand	200	225	275	300	

By solving the problem using the IZPM, at step 7 (d) we have to select the candidate cell for allocation. If cell (4, 2) is selected as presented in Table 4, then this selection leads to the optimal transportation plan presented in Table 5 with the total cost = $75 * 11 + 175 * 13 + 125 * 16 + 175 * 10 + 50 * 0 + 275 * 13 + 50 * 17 + 50 * 17 + 75 * 0 = 12525$. If cell (2, 2) is selected as presented in Table 6 it is not possible to satisfy the demand d_4 (100) although we have still $s_1 = 50$ and $s_4 = 50$ as shown in Table 5 and we cannot allocate them to cells (1, 4) and (4, 4) because there reduced costs are not zeros, so there is no solution in this case. Therefore, we can conclude that the arbitrary selection criterion is not valid all the time. In the next section the proposed method will be presented in order to avoid such drawback.

Table 4

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	11 0	13 0	17 9	14 9	0 5	250
s_2	16 0	18 0	14 1	10 0	0 0	350
s_3	21 5	24 6	13 225	10 0	0 0	125
s_4	15 0	17 50	21 0	18 0	0 1	0
s_5	21 5	23 5	19 6	15 5	0 75	0
Demand	200	175	0	300	50	1125

2. Drawbacks of the IZPM

Although the ZPM and the IZPM have been developed to find an optimal solution for the TP, both methods have some drawbacks in reaching such optimal solution. The situations of such drawbacks can be described as follows:

The first drawback exists in optimality condition of steps 4 and 4 (b) of the ZPM and the IZPM, where in some situation the optimality condition is satisfied but the supply (demand) cannot satisfy the sum of the demands (supplies).

In order to illustrate such case consider the balanced transportation problem as presented in Table 1, where s_i represents the supply of the source i and d_j represents the demand of the destination j .

Table 1

	d_1	d_2	d_3	d_4	Supply
s_1	2	2	2	4	1000
s_2	4	6	4	3	700
s_3	3	2	1	0	900
Demand	900	800	500	400	2600

By solving the problem using the IZPM, at step 4 (b) although each column demand is less than the supply s_1 , but supply s_1 cannot satisfy the total columns demands d_1 , d_2 and d_3 ($900 + 800 + 500 = 2200$) is greater than the supply s_1 (1000), because there is not another zero cells for allocation in the same column demands (d_1 , d_2 and d_3) as presented in Table 2.

Table 2

	d_1	d_2	d_3	d_4	Supply
s_1	0	0	0	2	1000
s_2	1	3	1	0	
s_3	3	2	1	0	
Demand	900	800	500	400	2600

So there is no solution in this case. Therefore, we can conclude that the optimality condition is not valid all the time. In the next section the proposed algorithm will be presented in order to avoid such drawback.

The second drawback exists in the arbitrary choices in steps 8 and 7 (d) of the ZPM and the IZPM where not every choice leads to an optimal solution.

A Modified Version of the Improved Zero Point Method for Solving Transportation Problems

Abo Shady, A. Mona¹, Ragab, M. Aly², El-Sherbiny, M. Mahmoud²

Abstract

The Zero Point Method (ZPM) was first introduced by Pandian and Natarajan [1] for solving the transportation problem. Samuel [2] has proposed the Improved Zero Point Method (IZPM) to justify the optimality condition in the ZPM. In this paper a modified version is proposed in order to handle the drawbacks in both methods. A comparison between the proposed algorithm and the ZPM and the IZPM is presented.

Key words: Optimal solution, Transportation problem, Zero point method.

1. Introduction

The transportation model is a special case of the linear programming problem (see for example [3], [4], [5] and [6]). It deals with transporting certain product from m sources to n destinations. The sources are production facilities with respective capacities $a_1, a_2, \dots, a_m$ and the destinations are warehouses with required levels of demands $b_1, b_2, \dots, b_n$. The penalty for transporting a unit from the source i to destination j of the given product is c_{ij} and the problem is to determine the amounts to be transported from all sources to all destinations such that the total transportation cost is minimized.

In 2010, the ZPM was first introduced by Pandian and Natarajan [1] as a method for solving the transportation problem to obtain an optimal solution.

The rest of the paper is organized as follows: in section 2, we describe the drawbacks of the IZPM. In section 3, we present the proposed algorithm. In section 4, illustrative examples is solved. Section 5 present a comparison between the ZPM , the IZPM and the proposed algorithm. Finally, section 6, presents the conclusion of the work.

¹ Egyptian Petroleum Research Institute, Nasr City, Cairo, Egypt.

² Department of Operations Research, Institute of Statistical Studies and Research, Cairo University, Giza, Egypt.

- [14] Adlakha V. and kowalski K., "A Quick Sufficient Solution to the More- for-Less Paradox in the Transportation Problem", Omega International Journal of Management Science, Vol. 26, Issue No. 4, 1998, pp. 541-547.
- [15] Reeb J. and Leavengood S., "Transportation Problem: A Special Case for Linear Programming Problems", Operations Research EM 8779, 2002, pp. 1-36.
- [16] Prakash S., "UNIT-3: Transportation Problem", Visvesvaraya Technological University, e-Learning Centre, Mysore, 06IP/IM74 Operations Research, 2013, pp. 1-39.
- [17] Fogiel M., The Operations Research Problem Solver, Research and Education Association, New York, N. Y. 10018, 1987.

6. References

- [1] F. L. Hitchcock (1941): "The distribution of product from several source to numerous localities", J. Maths. Phy. , 20, 224-230.
- [2] Yang I. and Liu L., "Fuzzy Fixed Charge Solid Transportation Problem and Algorithm", Applied Soft Computing, Vol. 7, 2007, pp. 879-889.
- [3] Haley, K.B., 1963, "The multi-index transportation problem", Oper. Res., 11, pp. 368-379.
- [4] Ojha A., Das B., Mondal S. and Maiti M., "A Solid Transportation Problem for an Item with Fixed Charge, Vehicle Cost and Price Discounted Varying Charge Using Genetic Algorithm", Applied Soft Computing, 2010, pp. 100-110.
- [5] Taha H., Operations Research an Introduction, Macmillan Publishing Co., Inc., 1982.
- [6] Charnes A., Cooper W. and Henderson A., an Introduction to Linear Programming, Wiley, New York, 1953.
- [7] Pandian P. and Natarajan G., "A New Algorithm for Finding a Fuzzy Optimal Solution for Fuzzy Transportation Problems", Applied Mathematical Sciences, Vol. 4, Issue No. 2, 2010, pp. 79 – 90.
- [8] Samuel A., "Improved Zero Point Method (IZPM) for the Transportation Problems", Applied Mathematical Sciences, Vol. 6, Issue No. 109, 2012, pp. 5421 – 5426.
- [9] Quddoos A., Javaid S. and Khalid M., "A New Method for Finding an Optimal Solution for Transportation Problems", International Journal on Computer Science and Engineering, Vol. No. 4, Issue No. 7, 2012, pp. 1271-1274.
- [10] Abo Shady M., Ragab A. and El-Sherbiny M., "A Modified Version of the Improved Zero Point Method for Solving the Transportation Problem ", Under Publishing.
- [11] Hillier F. and Lieberman G., Introduction to Operations Research, McGraw-Hill companies, Inc., 2001.
- [12] Pandian P. and Natarajan G., "A New Method for Finding an Optimal Solution of Fully Interval Integer Transportation Problems", Applied Mathematical Sciences, Vol. 4, Issu No. 37, 2010, PP. 1819-1830.
- [13] Sharma G., Abbas S. and Gupta V., "Optimum Solution of Transportation Problem with the Help of Zero Point Method", International Journal of Engineering Research & Technology, Vol. 1, Issue 5, 2012, pp. 1-6.

Table 14

	d_1	d_2	d_3	d_4	d_5	
s_1	11 (6) 200 0	13 (7) 50 0	17 9	14 9	0 5	250
s_2	16 0	18 (8) 125 0	14 1	10 (5) 175 0	0 (3) 50 0	350
s_3	21 5	24 6	13 (1) 275 0	10 (4) 125 0	0 0	400
s_4	15 0	17 (9) 50 0	21 9	18 9	0 1	50
s_5	21 5	23 5	19 6	15 5	0 (2) 75 0	75
Demand	200	225	275	300	125	1125

An alternative optimal allocation of such example is: $x_{11} = 200$, $x_{12} = 50$, $x_{22} = 125$, $x_{24} = 175$, $x_{25} = 50$, $x_{33} = 275$, $x_{34} = 125$, $x_{42} = 50$ and $x_{55} = 75$ with the total cost = 12525.

4. Comparison between the proposed algorithm and the ZPM, the IZPM and the ASM method

We have chosen 40 transportation problems from different references as follows: one problem from Taha [5], one problem from Hillier and Lieberman [11], one problem from Qudus et al. [9], four problems from Pandian and Natarajan [12], two problems from Sharma et al. [13], two problems from Adlakha and Kowalski [14], four problems from Reeb and Leavengood [15], twelve problems from Prakash [16], six problems from Fogiel [17] and eight problems from other references.

We solved these 40 problems using the proposed algorithm and compared the solutions with the ZPM, the IZPM and the ASM method. The conclusion was that using our algorithm gives the optimal solution for the 40 problems. Solving the same problems using the ASM method gives the optimal solution for 39 problems, while solving the same problems using the ZPM and the IZPM gives optimal solution for 34 problems only.

5. Conclusion

In this paper an algorithm for solving the transportation problem has been proposed using two stages: the first stage depends on a modified version of the Hungarian algorithm to create a reduced cost transportation matrix. In the second stage, the allocation depends on the zero cells in the reduced cost transportation matrix, with the minimum cost in the original transportation cost matrix. The proposed algorithm gets an optimal solution for the transportation problems compared with the ZPM, the IZPM and the ASM method.

Table 12

	d_1	d_2	d_3	d_4	d_5	
F1	11 200 0	13 50 0	17 9	14 9	0 5	250
F2	16 0	18 125 0	14 1	10 225 0	0 0	350
F3	21 5	24 6	13 275 0	10 75 0	0 50 0	400
F4	15 0	17 50 0	21 9	18 9	0 1	50
F5	21 5	23 5	19 6	15 5	0 75 0	75
Demand	200	225	275	300	125	1125

The optimal allocation of such example is: $x_{11} = 200$, $x_{12} = 50$, $x_{22} = 125$, $x_{24} = 225$, $x_{33} = 275$, $x_{34} = 75$, $x_{35} = 50$, $x_{42} = 50$ and $x_{55} = 75$ with the total cost = 12525.

In the proposed algorithm, in step 8 the cells (2, 5) and (3, 5) have the same transportation costs. If we choose cell (2, 5) for allocation as presented in Table 13, instead of cell (3, 5) in Table 11, we get an alternative solution in Table 14.

Table 13

	d_1	d_2	d_3	d_4	d_5	
S_1	11 0	13 0	17 9	14 9	0 5	250
S_2	16 0	18 0	14 1	10 0	0 50 0	300
S_3	21 5	24 6	13 275 0	10 0	0 0	125
S_4	15 0	17 0	21 9	18 9	0 1	50
S_5	21 5	23 5	19 6	15 5	0 75 0	0
Demand	200	225	0	300	0	1125

Table 10

	d_1	d_2	d_3	d_4	d_5	
s_1	11 0	13 0	17 9	14 9	0 5	250
s_2	16 0	18 0	14 1	10 0	0 0	350
s_3	21 5	24 6	13 25	10 0	0 0	125
s_4	15 0	17 0	21 9	18 9	0 1	50
s_5	21 5	23 5	19 6	15 5	0 75	0
Demand	200	225	0	300	50	1125

Step 8: Since there are two zero cells having the minimum transportation cost (0) in the original transportation cost table (Table 1), then choose cell (3, 5) and allocate it with 50 as presented in Table 11.

Table 11

	d_1	d_2	d_3	d_4	d_5	
s_1	11 0	13 0	17 9	14 9	0 5	250
s_2	16 0	18 0	14 1	10 0	0 0	350
s_3	21 5	24 6	13 25	10 0	0 50	75
s_4	15 0	17 0	21 9	18 9	0 1	50
s_5	21 5	23 5	19 6	15 5	0 75	0
Demand	200	225	0	300	0	1125

Step 9: Since all supplies and demands are not satisfied, repeat steps from 7 to 9 until all supplies and demands are satisfied.

Step 10: Table 12 giving the optimal transportation plan.

Table 8

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	8	8	4	250
s_2	1	①	1	0	0	350
s_3	6	7	0	0	0	400
s_4	0	0	8	3	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Repeating Step 6: Subtracting the smallest cost (1) in the uncovered entries in Table 8, and adding it to the intersection of every two lines, we get Table 9.

Table 9

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	9	9	5	250
s_2	0	0	1	0	0	350
s_3	5	6	0	0	0	400
s_4	0	0	9	9	1	50
s_5	5	5	6	5	0	75
Demand	200	225	275	300	125	

Repeating step 4: Checking the first row, the zeros in that row of destinations d_1 , d_2 and d_5 with total demands = $200 + 225 + 125 = 550$ is greater than the supply s_1 (250). Also, if the same check is repeated for all the rows and all the columns the allocation condition will be satisfied, so, go to step 7.

Step 7: Since there is a single zero cell (5, 5) in row s_5 and a single zero cell (3, 3) in column d_3 , then allocate both cells as presented in Table 10.

Table 5

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	4	4	0	250
s_2	5	5	1	0	0	350
s_3	10	11	0	0	0	400
s_4	4	4	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Step 6: Subtracting the smallest cost (4) in the uncovered entries in Table 5, and adding it to the intersection of every two lines, we get Table 6.

Table 6

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	8	8	4	250
s_2	1	1	1	0	0	350
s_3	6	7	0	0	0	400
s_4	0	0	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Repeating step 4: Checking the first row, we notice that the zeros in that row of destinations d_1 , d_2 and d_5 with total demands = $200 + 225 + 125 = 550$ is greater than the supply s_1 (250). Also, if the same check is repeated for all the rows and all the columns, we conclude that d_1 and d_2 in Table 6 are not satisfying the condition as illustrated in Table 7.

Table 7

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	8	8	4	250
s_2	1	1	1	0	0	350
s_3	6	7	0	0	0	400
s_4	0	0	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Repeating Step 5: Covering all zeros in Table 7 with the minimum number of horizontal and vertical lines and keeping the unsatisfied columns (d_1 and d_2) uncovered gives Table 8.

Step 3: Subtracting the minimum cost of each column from every entry in that column, we get Table 3.

Table 3

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	4	4	0	250
s_2	5	5	1	0	0	350
s_3	10	11	0	0	0	400
s_4	4	4	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Step 4: Checking the first row, we notice that the zeros in that row of destinations d_1 , d_2 and d_5 with total demands = $200 + 225 + 125 = 550$ is greater than the supply s_1 (250). Doing the same check to the rest of the rows and the columns, we conclude that d_1 and d_2 in Table 11 are not satisfying the condition as illustrated in Table 4.

Table 4

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	0	0	4	4	0	250
s_2	5	5	1	0	0	350
s_3	10	11	0	0	0	400
s_4	4	4	8	8	0	50
s_5	10	10	6	5	0	75
Demand	200	225	275	300	125	

Step 5: Covering all zeros in Table 4 with the minimum number of horizontal and vertical lines and keeping the unsatisfied columns (d_1 and d_2) uncovered gives Table 5.

it to the intersection of every two lines and go to step 4.

Step 7: If there is a single zero in any column or / and row, then allocate it.

Step 8: From all the zero cells in the reduced cost transportation matrix, choose the cell with the minimum transportation cost in the original transportation cost matrix and allocate it, if there is a tie, then choose any one.

Step 9: If all supplies and demands are not satisfied, go to step 7, otherwise go to step 10.

Step 10: The last table gives the optimal solution.

3. Example

Consider the following unbalanced transportation problem as presented in Table 1, where s_i represents the supply of the source i and d_j represents the demand of the destination j .

Table 1

	d_1	d_2	d_3	d_4	Supply
s_1	11	13	17	14	250
s_2	16	18	14	10	350
s_3	21	24	13	10	400
s_4	15	17	21	18	50
s_5	21	23	19	15	75
Demand	200	225	275	300	

Step 1: Since the problem is unbalanced, adding the dummy destination d_5 with demand = 125, we get Table 2.

Table 2

	d_1	d_2	d_3	d_4	d_5	Supply
s_1	11	13	17	14	0	250
s_2	16	18	14	10	0	350
s_3	21	24	13	10	0	400
s_4	15	17	21	18	0	50
s_5	21	23	19	15	0	75
Demand	200	225	275	300	125	

Step 2: Since the minimum cost in every row is zero, we still have Table 2.

drawbacks in the ZPM and the IZPM due to the arbitrary selection criterion in both methods by proposing a new criterion for the selection of the zero cells depending on the original transportation cost matrix. This algorithm also adds a new condition to adjust the optimality criteria in both methods.

This paper introduces an algorithm based on the Hungarian algorithm for solving the TP. The rest of the paper is organized as follows: in section 2, the proposed algorithm is presented. In section 3, an illustrative example is solved. Section 4 presents the comparison between the proposed algorithm and the ZPM, the IZPM and the ASM methods. Finally, section 5 presents the conclusion of the work.

2. The proposed Algorithm

The proposed algorithm is applied in two stages. The first stage consists of six steps depending on a modified version of the Hungarian algorithm [5] to create a reduced cost transportation matrix. The second stage consists of four steps where the allocation depends on the zero cells in the reduced cost transportation matrix, which have the minimum cost in the original transportation cost matrix. The steps of the algorithm are illustrated as follows:

- Step 1:** If the TP is unbalanced change it into a balanced one by adding a dummy row or column.
- Step 2:** Subtract the minimum cost of each row from all the entries of that row.
- Step 3:** Subtract the minimum cost of each column resulting from step 2 from all the entries of that column.
- Step 4:** If each row supply (column demand) is less than or equal to the sum of the demands (supplies) whose reduced costs are zeros in that row (column) and in the same time the sum of the row supplies (column demand) in these rows (columns) is less than or equal to the demand of that column (the supply of that row), then be sure there are another zero cells in the same row supplies (column demands), then go to step 7, otherwise, go to step 5.
- Step 5:** Cover all zeros of the reduced cost transportation table resulting from step 3 with the minimum number of horizontal and vertical lines and keep the unsatisfied rows or/and columns from step 4 uncovered.
- Step 6:** Subtract the smallest cost in the uncovered entries from each uncovered entries of the reduced cost transportation matrix, add

Solving the Transportation Problems Using the Hungarian Algorithm

Abo Shady, A. Mona ¹, Ragab, M. Aly ², El-Sherbiny, M. Mahmoud ²

Abstract

In this paper an algorithm for solving the Transportation Problem (TP) is proposed to obtain the optimal solution directly using the Hungarian algorithm. The results of this algorithm are compared with the Zero Point Method (ZPM), the Improved Zero Point Method (IZPM) and the ASM method.

Key words: Hungarian algorithm; Optimal solution; Transportation problem; Zero point method; Improved zero point method; ASM method.

1. Introduction

The transportation problem (TP) is considered to be a special class of the linear programming problem. It was firstly presented by Hitchcock [1], where sources and destinations constraints are taken into consideration. In the real life, other constraints can be considered, such as product type constraints or transportation mode constraints [2]. For such cases, the TP can be turned into a Solid Transportation Problem (STP) by adding the conveyances capacities constraints [3]. The STP degenerates into the TP as the number of conveyance is only one [4].

Three different methods provide the initial basic feasible solution (IBFS) for the TP; such as northwest-corner, least-cost method and Vogel's approximation method. The IBFS is improved using the modified distribution method [5] or the stepping stone method [6] to reach an optimal solution.

Recently, the ZPM [7], the IZPM [8], and the ASM method [9] have been proposed for obtaining the optimal solution directly for the TP. Such methods require the least number of iterations to reach optimality compared with finding IBFS and improving it. Some drawbacks exist in these methods. Abo Shady et al. [10] proposed an algorithm to handle the

¹ Egyptian Petroleum Research Institute, Nasr City, Cairo, Egypt.

² Department of Operations Research, Institute of Statistical Studies and Research, Cairo University, Giza, Egypt.

- Kerzner, H. (2003). Project management: a systems approach to planning, scheduling, and controlling. 8th Ed., Wiley, Hoboken, NJ.
- Liu, L., S.A. Burns, and C.W. Feng., (1995), "*Construction time-cost tradeoff analysis using LP/IP hybrid method*", Journal of Construction Engineering and Management, Vol. 121, No.(4), pp. 446-454.
- Li, H. and Love, P., (1997), "*Using Improved Genetic Algorithms to Facilitate Cost Optimization*", Journal of Construction Engineering and Management, Vol. 13, No. (3), pp. 233-237.
- Lu, M., & AbouRizk, S. M. (2000), "*Simplified CPM/PERT simulation model*", Journal of Construction Engineering and Management, Vol. 126, No. (3), pp. 219-226.
- Lee, D. E. (2005), "*Probability of Project Completion Using Stochastic Project Scheduling Simulation*", Journal of Construction Engineering and Management, Vol. 13, No. (3), pp. 310-318.
- Michael, E., Radhamés, A., (2008), "*A dynamic simulation-based crashing method*", *Proceedings 2008 of the Winter Simulation Conference, Rochester, NY*, pp. 2370-2376.
- Michael, E., Radhamés A. (2009), "*Project Management: A Simulation-based Optimization Method for Dynamic Time –Cost Tradeoff Decisions*", M.S. thesis, New York University, Kate Gleason, New York.
- Pritsker, A. A. B. (1986). *Introduction to Simulation and SLAM II. 3rd Ed., Wiley & Sons, Inc, New York.*
- Surabaya, y.A., Ladamay, S. A., (2011) "*Project Rellability: Probability of a project meets its Quality-Cost-Time Target under uncertainty*", International Journal of Electronic Business Management, Vol. 9, No. (3), pp. 220-230.
- Tayfur, A., and Belamed, M., (2010)., "*Simulation Modeling and Analysis with ARENA*", Elsevier Inc, USA.
- Van Slyke, R. M., (1966), "*Monte Carlo Methods and the PERT Problem*", Operations Research, Vol.11, No. (5), pp. 839-860.

Table 6: Configuration of Alternatives Resources from Original, DSA and SDPMO Method.

		Activity 1	Activity 2	Activity 3	Activity 4
configuration of renewable resources	original	1	1	1	1
	DSA	1	1	2	2
	SDPMO	1	1	2	1

6-Conclusion

This research provides SDPMO method to solve the problems of decision-making under uncertainty managed projects. This method is designed to minimize the average total cost of projects by optimize utilization of available resources. In the above example, there was a reduction in the cost of the project to 38.19 by the DSA method and to 25 by the SDPMO method whereas the increase happened in completion time of project from 55.97 of the DSA. method to 58.62 by the SDPMO method, but without lateness penalties cost.

Recommendations for Future Research

- 1• Supported Problems with multi objectives function.
- 2• Supported non linear problem.
- 3• Find a solution to this problem by using the methods of artificial intelligence and compare it with the current method.
- 4• Develop the proposed method to be applied on the concurrently managed projects.
- 5• Develop the proposed method to take into account the cash flows.

References

- Ammar, M. A. (2011),” *Optimization of project time-cost trade-off problem with discounted cash flows*”, Journal of Construction Engineering and Management, Vol. 137, No. (1), pp. 65–71.
- Bissiri, Y., and Dunbar, S. (1999), "Resource allocation model for a fast-tracked project". International Conference 1999 on Intelligent Processing and Manufacturing of Materials, pp. 635- 640.
- Hillier, F. S., and G. J. Lieberman. (2001). *Introduction to Operations Research. 7th Ed., McGraw-Hill, New York.*
- Haga, W. A., and Marold, K. A. (2004). "A simulation approach to the PERT CPM time-cost tradeoff problem", Project Management Journal, Vol. 35(2), pp. 31-37.

number 4 will use the resources number 2 and activity number 1 will use the resources number 1.

Table 4: Average Start, Duration and Completion Time for all activities.

		Average	Variance	S.d	C.I 95%
activity 1	Start time	0	0	0	0-0
	Duration	15.0227	2.08769	1.44488	12.1907- 17.8547
	Complete time	15.0227	2.08769	1.44488	12.1907- 17.8547
activity 2	Start time	15.0227	2.08769	1.44488	12.1907- 17.8547
	Duration	19.5486	10.1191	3.18106	13.3138- 25.7835
	Complete time	34.5713	12.2438	3.49912	27.713- 41.4296
activity 3	Start time	34.5713	12.2438	3.49912	27.713- 41.4296
	Duration	7.98923	0.107429	0.327764	7.34681- 8.6316
	Complete time	42.5605	12.2361	3.49801	35.7044-49.4166
activity 4	Start time	42.5605	12.2361	3.49801	35.7044- 49.4166
	Duration	16.0653	2.89762	1.70224	12.7289-19.4017
	Complete time	58.6258	12.572	3.5457	51.6762- 65.5754

Table 5: Summarized Comparison between Original, DSA and SDPMO Method.

	Duration		Cost	
	Average	Std. Dev.	Average	Std. Dev.
Original	70.97	5.29	631.10	473.09
DSA	55.97	4.66	38.19	26.49
SDPMO	58.62	7.09	25	14.03

The above table shows the solution for the presented example from three different methods, the duration of project by original method is 70 units with std.dev 5.29 and total cost of project is 631.10 units with std.dev 473.09, the duration of project by DSA method is 55.97 units with std.dev 4.66 and total cost of project is 38.19 units with std.dev 26.49, the duration of project by original method is 58.62 units with std.dev 4.66 and the total cost of project is 25 units with std.dev 14.03, the suggested configuration of resources from three methods is shows in table 6.

The following a prototype functions to generate Activity duration from above the probability distribution (Tayfur and Belamed, 2010):

1- A prototype triangular function:

Generate random number between (0:1), If (random number < (mod- min) / (max - min)) then Activity duration = min + sqrt ((mod - min) * (max- min) * random number between (0:1)), otherwise Activity duration = max – sqrt ((max - mod) * (max – min)*(1- random number between (0:1)))).

2- A prototype uniform function:

Activity duration= (min) + (max - min) * random number between (0:1).

3- A prototype exponential function:

Activity duration= -mean* LN random number between (0:1).

In the above table the cost for alternative number 1 is zero because the activity must be completed using the current available resources (base alternative). For each one of the activities there are 2 available resources that can be hired to complete the work for an additional cost. In order to facilitate the application of the SDPMO optimization method to real project, engine program, simulators are fully integrated. The following output reports for the above example which includes the important data for project as Table 2. Average completion time and average cost for project, in Table 3. Configuration of minimize the average total cost, in Table 4. Start time, finish time for all activities, in Table 5. Is summarized comparison between Original, DSA and SDPMO method, in Table 6. Configuration of alternatives resources from original, DSA and SDPMO method.

Table 2: Average Completion Time and Average Cost for Project.

	Average	Variance	S.d	C.I 95%
Completion time	58.6258	50.2881	7.09141	44.72 - 72.52
completion cost	25	196.922	14.0329	-2.50447 – 52.5045

Table 3: Configuration of Minimize the Average Total Cost.

activity	1	2	3	4
resource	1	1	2	1

The above table shows the configuration of resources to implement the project activities as follows, activity number 1 will use the resources number 1, activity number 2 will use the resources number 1, activity number 3 will use the resources number 2, activity

5- Example

The following example on SDPMO, sample project has 4 activities is described in following Table1. The activities that make up project may have several alternatives each with an associated cost and stochastic duration. The alternatives of an activity might include working over time, bringing in additional resources, or subcontracting the work. The alternative potential of an activity is the number of alternatives associated with the activity. The costs associated with implementing the alternatives of an activity can be independent from each other. For the first activity in the example, the alternative potential is assumed to be 1, because the activity must be completed using the current available resources (this alternative is referred to as base alternative). For each one of the remainder activities, there are 2 contractors available that can be hired to complete the work for an additional cost. If the project is late, a lateness penalty is assessed as a linear function of the quantity of time by which the project is late. For example, a project that is due at time 65 days with a lateness penalty of 100 per time unit.

Table 1: SDPMO Sample Project Specifications.

Activity	Alternative			Additional cost		predecessors
	1(Base)	2	3	2	3	
1	T10 15 20			-	-	-
2	T10 20 30	U15 30	U10 20	10	30	1
3	T15 20 25	T7 8 9	E13	25	15	2
4	U13 19	T11 13 15	U10 28	10	10	3

The activity time can be estimated using any probability distribution in this paper only the following probability distributions are considered uniform, triangular and exponential distribution, i.e. the stochastic duration for any resource might be represented by the following formats:

- Triangular distribution: T min mode max, e.g. T10 15 20;
- Uniform distribution: U min max, e.g. U15 30;
- Exponential: E mean, e.g. 13;

4.3 - Simulation model

The simulation model is used to evaluate the new configuration to obtain the optimal configuration alternatives that minimize the average total cost. The simulation model is runs multi replication to get rid of the uncertainty associated with the activity duration. If average total cost for new configuration less than old average total cost for old configuration replace the old configuration with new configuration, else if number point of solution = stopping criteria then print the outputs, else go to the engine. For each replication of the simulation model instance of the following steps is applied:

- Step1: Repeat from activity 1 to activity N, $j=1, \dots, N$, N represented the number of activity in project.
- Step 2: Set the start time and completion time for activity $j = 0$;
- Step 3: Set index $k = 1$;
- Step 4: If activity j is not completed and the all previous activities of activity j is completed go to step 5, otherwise go to step 6.
- Step5: If activity k before activity j , start time for activity $j = \max$ (start time for activity j , completion time for activity k).
- Step 6: If $k =$ the numbers of activities in project then go to step 7, otherwise set $k = k+1$ and go to step 5.
- Step 7: Generate sample time for activity j from probability distribution.
- Step 8: Set the completion time for activity $j =$ the start time for activity $j +$ sample time.
- Step 9: Increment the number of activates is completed +1.
- Step 10: If the number of activity completed = the number of activities in project, go to step 12 otherwise $j= j+1$ and go to step 3.
- Step11: Set the total completion time of project = max (completion time for activates project).
- Step12: Set the total cost for project = Penalty function (target time for project total completion time for project) + alternatives resources cost
- Step13: Store the total completion time and total cost after any replication to estimate the average after multi replication.

4.4- Output data

The output data includes the average total cost, average completion time for projects, average start and finish time for all activities and average duration for all activities.

4- Methodology

The purpose of this research is to develop a dynamic simulation method capable of evaluating the project network to obtain the configuration renewable resources that minimize the average cost for project. This research introduces a simulation of dynamic project management optimization (SDPMO) method to obtain the optimal configuration resources by interaction between the following system components:

4.1- Input data

The input data includes information about the number of activities for project, predecessors, and the number of alternatives resources of each activity in project, with their associated stochastic durations and costs, target completion time and the penalty cost due to lateness.

4.2- Engine

The engine steps to generate the configuration from resources can be illustrated in the following.

- Step 1: Specify the range of alternatives renewable resources for all activities in project.
- Step 2: Connect the system by time (fast variable) with random number to obtain all configuration alternatives without redundancy.
- Step 3: Generate alternatives resources for all activities from the range of possible alternatives renewable resources for all activities by using the equation $\text{range} = \text{number of available alternatives}$; $\text{new configuration} = (\text{rand}() / 100 \% \text{ range} + \text{min})$ to obtain the configurations alternatives resources.
- Step 4: Transform the new configuration of alternatives renewable resources to ASCII code (American Standard Code for Information Interchange) and concatenate it, if new configuration is first configuration save it and go to the following step, Else compare between all saved by ASCII code for configuration alternatives renewable resources previously stored if found then generate the new configuration, else go to the next steps.
- Step 5: Set the number of points in solution + 1.
- Step 6: Calculate the cost for new configuration alternatives resources if sum cost for new configuration alternatives renewable resources less than evaluated for old configuration alternatives send the configuration alternatives renewable resources to simulation model for evaluation, else generate the new configuration.

uncertainty is designed to solve the traditionally time-cost tradeoff problem, the activities durations of this method are represented by beta distribution only. (Michael and Radhamés, 2009) proposed A dynamic selection of activity alternatives (DSA) method to obtain the optimal configuration of alternatives under uncertainty, they are designed to solve the generalized stochastic time-cost tradeoff problem. The method considers that there are different alternatives associated with each activity, each one has its own distribution of completion time and associated cost. DSA is a heuristic method because the optimization Industrial Strength Compass (ISC) engine used is not designed to guarantee optimality for these cases because the alternative is non linear cost. If an optimization engine can be developed for these cases the solutions provided can be optimal. This research develops optimization engine to guarantee the optimality solution. (Ammar 2011) introduced optimization of project time-cost trade-off (TCTO) problem with discounted cash flows (DCF). (Ammar 2011) proposed a nonlinear mathematical optimization model for project TCTO problem which minimizes project direct cost and takes into account discounted cash flows. Surabaya et al (2011) proposed A Monte Carlo Simulation and Generalized Activity Network utilized method to evaluate probability of a project whether it met its Quality-Cost-Time target under uncertainty.

3- Problem Formulation

The problem can be illustrated by the following mathematical model.

Minimize $Z = C$. C = average cost from resources for all activity and the penalty cost for project.

Subject to:

$T_j - T_i \leq t_{ij}$ for all i and j where $j > i$

$t_{ij} \geq 0$ where:

T_i = earliest event time for node i .

T_j = earliest event time for node j .

t_{ij} = duration for activity (i, j).

$R_i \leq k$ k = the number of available resources for all activity i .

$K > 0$ and integer.

simulation model that evaluates "criticality indexes" which are the probability of various activities being project networks to provide a histogram of the project completion time distribution, which can be used to perform risk analysis. (Liu, et al.1995) introduced the time cost optimization model using integer programming / linear programming. It is assumed that an activity can be completed in various durations with associated cost. Duration of various activities is considered to be discrete for a project. (Li and Love, 1997) formulated a different type of time cost tradeoff problem that occurs when the project is running behind schedule. To meet the deadline, the project duration should be reduced to a certain limit. It is assumed that not all the activities are crash. The objective is to find which activities should be crashed and by what duration to shorten the total duration to the targeted limit with minimum compressing cost. (Bissiri and Dunbar, 1999) proposed a method to obtain the average time of each activity to crash a project network by use simulation, the critical path, and the near critical paths. A near critical path in this model is a path which length is smaller than the original completion date but it is larger than the target completion date. (Lu and AbouRizk, 2000) introduced the discrete event modeling approach to CPM and PERT by simulation model to a simplified critical activity. (Hillier and Lieberman, 2001) introduced the original versions of CPM and PERT have some significant differences, with time they have been considered as one technique called CPM and PERT. The PERT method considers the mean and variance of each activity to describe its duration and to represent the uncertainty associated with it. (Haga and Marold, 2004) proposed a simulation-based method that deals with the time cost tradeoff involved with crashing a project. The complete distribution of project completion time needs to be considered when crashing". The method that they proposed is two steps. The first step is to apply the traditional PERT method to crash the project, and the second step consists in testing each activity that had not been crashed to the upper crashing limit to determine if crashing that activity further reduces the average total cost of the project. That can increase the cost of the project, which are crashing costs and overrun costs. (Lee, 2005) presented a software tool, SPSS, which can be used to determine the probability associated with the completion of the project by a target date specified. (Haga and Marold, 2005) provided a dynamic phase (throughout the project execution) to obtain the heuristic solution. (Michael and Radhamés, 2008) proposed A dynamic crashing method for project management using simulation based optimization to obtain the optimal crashing configurations under

The tradeoff between the resources cost and the project cost (the project cost in this research is the sum of resources cost and penalty cost) is illustrated in Figure 1. For some combinations of resources, diminishing returns may be realized until a point where the total cost may begin to increase. That is, some combination of activity resources, although the combination would further reduce the completion time of the project, the cost of using these sets of resources would cost more than the penalty cost that would result from the project being late.

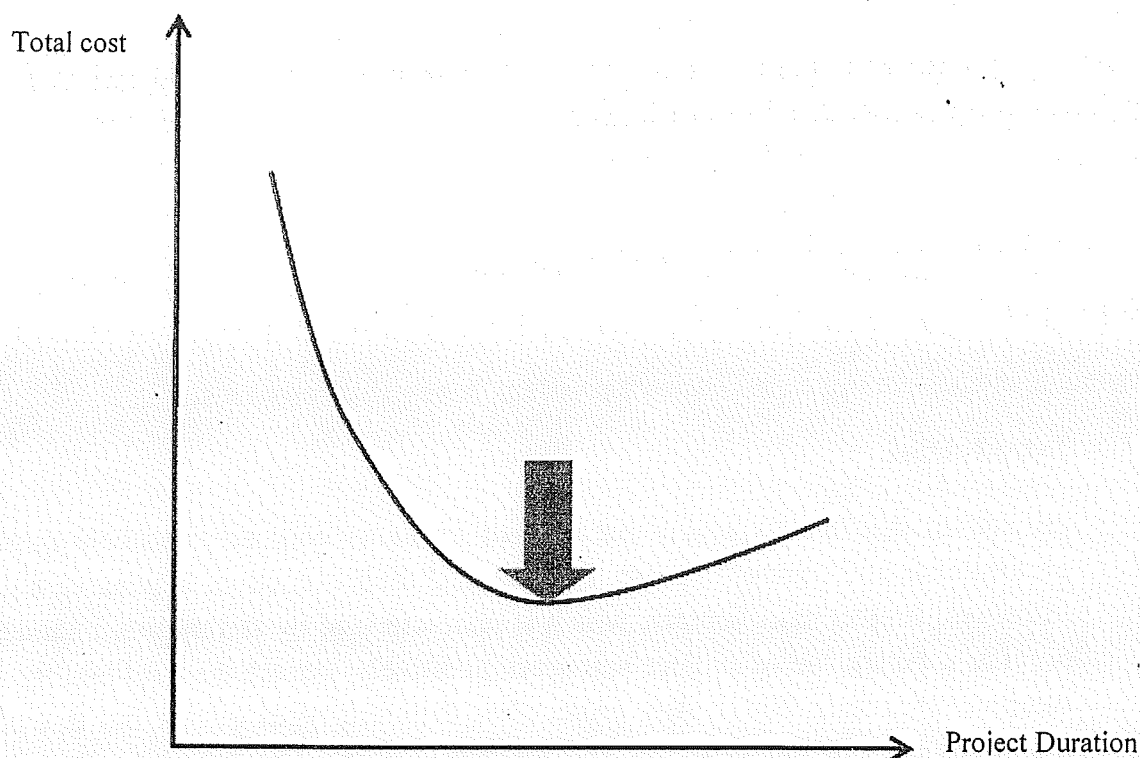


Figure 1: Relationship between Reduction in Project Duration and the Total Cost (resources cost + Penalty cost).

2- Related Work

A deterministic approach to calculate the duration of a project (CPM) and a probabilistic approach (PERT) are methods have been used since the 1950s to estimate the completion time of a project. (Van Slyke, 1966) introduced Monte Carlo simulations to study PERT. Van Slyke demonstrated several advantages of using simulation including more accurate estimates of the true project length, flexibility in selecting any distribution for activity times, and the ability to calculate "criticality indexes and critical path ". (Pritsker, 1986) presented the

Simulation of Dynamic Project Management Optimization

¹ M, M, Ghada., ² A, M, Atef., ³ G, H, Mohamed.,
⁴ E, H, EL Hamy.

Abstract

This paper introduces method to evaluate the project networks under uncertainty in activity time and determine the optimum configuration of alternatives renewable resources that minimizes the average cost. In an effort to optimize the project time-cost tradeoff, optimization method is presented. In particular, the method minimizes the expected project cost due to lateness penalties and the resources cost. The method is designed to be implemented in two cases. The first case, referred to as the static case, is implemented prior to the start of the project. The second case, referred to as the dynamic case, is implemented as the project progresses.

Keywords

Project management, time cost tradeoff problem, renewable resources.

1- Introduction

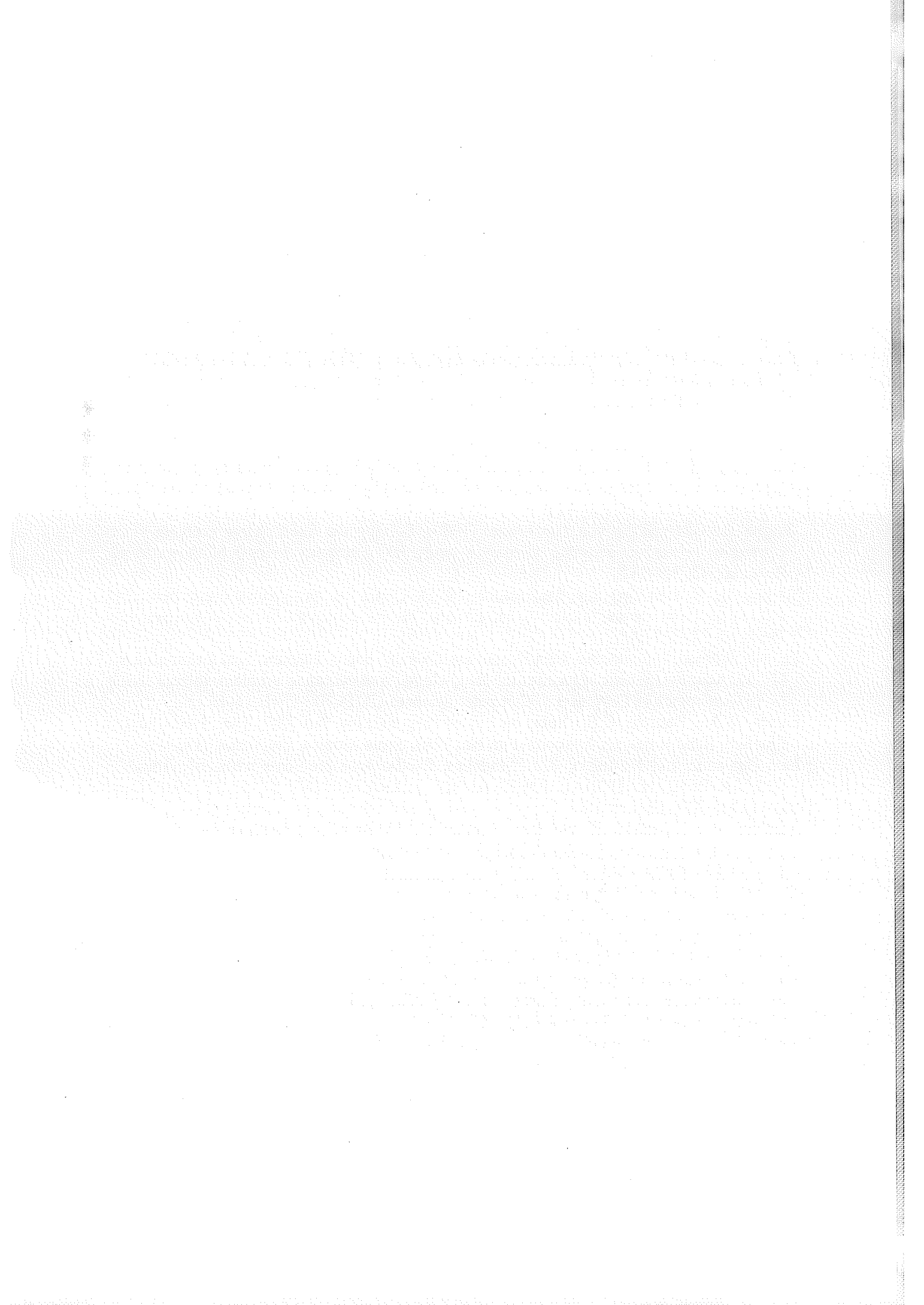
Project management is a tool that is used by many companies to help improve performance and competitiveness. Projects and their execution, in general, require resources. Project management, which is characterized by techniques, intended to provide a better use of project resources positively impact the profitability of a company (Kerzner, 2003). Any project has different types of risk but this paper will focus on schedule time risk and associated costs. The Schedule time risk is not completing project activities on time, resulting in a late completion of the project. Late project completion generally has negative effects for the company such as penalty costs and customer dissatisfaction. The goal of this paper is to address the time cost tradeoff problem respect to minimizes in expected total cost of project under uncertainty specifically; activities that make up a project may have several alternatives resources each with an associated cost and stochastic activity duration.

1-Assistant Lecturer, Department of Information Systems Obour High Institute for Management and computers and Information Systems

2- Professor, Department of Mathematical Statistics-ISSR-Cairo University

3- Professor of Industrial Engineering and Operations Research, Faculty of Engineering, Cairo University

4-DR, Akhbar AL Youm Newspaper



- Raidl, B. A., Julstrom, G.R., (2003). Edge-sets: An effective evolutionary coding of spanning trees. *IEEE Transactions on Evolutionary Computation*, 7(3), 225-239.
- Rousseau, J. M., (1973). A cutting plane method for the fixed cost problem, Doctoral dissertation, Massachusetts Institute of Technology. Cambridge, MA.
- Su, S., Zhan, D.C., (2006). New genetic algorithm for the fixed charge transportation problem, *Proceedings of the 6th World Congress on Intelligent Control and Automation*, 7039-7043.
- Sun, M., McKeown, P.G., (1993). Tabu search applied to the general fixed charge problem ,*Annals of Operations Research*, 41(14):405-420.
- Sun, M., Aronson, J.E. McKeown, P.G.,(1998). A tabu search heuristic procedure for the fixed charge transportation problem, *European Journal of Operational Research*,106 :441-456.
- Taguchi, G. (1986). Introduction to quality engineering. White Plains: Asian Productivity Organization/UNIPUB.
- Timmis, J., T Knight, L. N. Catro and Hart, (2004). An overview of artificial immune systems. *Computation in Cells and Tissues: Perspectives and Tools Thought*. Natural Computation Series, Springer-Verlag, 51-86.
- Tsai, J. T., Ho, W. H., Liu, T. K., & Chou, J. H. (2007). Improved immune algorithm for global numerical optimization and job shop scheduling problems. *Applied Mathematics and Computation*, 194, 406-424.
- Wright, D., C. Haehling von Lanzenauer (1989). Solving the fixed charge problem with lagrangian relaxation and cost allocation heuristics. *European Journal of Operational Research*, 42, 304-312.
- Wright, D., C. Haehling von Lanzenauer (1991). COLE: A new heuristic approach for solving the fixed charge problem Computational results. *European Journal of Operational Research*, 52, 235-246.
- Zalinda Othman, Mohammad-Reza Rostamian Delavar, Sarah Behnam, Sina Lessanibahri, (2011). Adaptive genetic algorithm for fixed-charge transportation problem, *International Multi Conference of Engineers & Computer Sciences (IMECS)* March, 16-18.
- Hirsch, W.M., Dantzig, G.B.: Notes on Linear Programming Part XIX, The Fixed Charge Problem, Rand Research, Memorandum No. 1383, Santa Monica, California. (1954)
- Lev, Benjamin and Kowalski, K . "Modeling fixed-charge problems with polynomials" *OMEGA INTERNATIONAL JOURNAL OF MANAGEMENT SCIENCE* 39.6 (Dec 2011):725-729
- Adlakha, V , Kowalski, K and Lev, Benjamin . "A Branching Method for the Fixed Charged Transportation Problem" *OMEGA, The International Journal of Management Sci* 38.5 (Oct 2010):393-397
- Kowalski, K and Lev, Benjamin. "On step fixed-charge transportation problem" *OMEGA, The International Journal of Management Sci* 36.5 (Oct 2008):913-917
- Kowalski, K and Lev, Benjamin . "New Approach to Fixed charges Problems (FCP)" *INT. J. OF MGMT. SCI. AND ENGINEERING MGMT.* 2.1 (Feb 2007):75-80
- Adlakha, V , Kowalski, K , Vemuganti, R. R. and Lev, Benjamin . "More-for-less algorithm for fixed-charge transportation problems" *OMEGA, THE International JOURNAL OF Management SCI* 35.1 (Jan 2007):116-127
- Sandrock K. A simple algorithm for solving small, fixed-charge transportation problem, *Journal of Operational Research Society* 1988:39(5):467-75
- Adlakha V, Kowalski, K. A simple heuristic for solving small fixed-charge transportation problems. *OMEGA-The International Journal of Management Science* 2003:31(3):205-11.

It can be observed that Table 13 provides an improved solution to the problem which meets the demand and supply constraints.

5. Conclusion

This paper has proposed a new version of SFCTP in the form of Multi Step Fixed Charge Transportation Problem (MSFCTP). One of its special versions – Multi Flat Step Fixed Charge Transportation Problem (MFSFCTP) considers the problem of minimizing transportation cost in situations where an opening cost is incurred for every route used in the solution and increases in multiples of shipped quantity, with the opening cost to remain flat up to a certain quantity and increase in multiples of this flat cost. The paper describes and presents the mathematical models of MSFCTP and MFSFCTP together with a heuristic algorithm for solving MFSFCTP. Two problems have been solved to evaluate and demonstrate the performance of the proposed algorithm.

References

- Altassan, Khalid M., El-Sherbiny, Mahmoud M., Ahmed D. Abid (2012), A Novel Artificial Immune Algorithm for Solving Fixed Charge Transportation Problems, submitted to 2012 International Conference on Innovation and Information Management.
- Balinski, M.L. (1961). Fixed cost transportation problem. *Naval Research Logistics Quarterly*, 8, 41-54.
- De Castro, L.N. and J. Timmis, (2002). An artificial immune network for multimodal function optimization. in *Proc. of IEEE congress on evolutionary computation*, Vol. 1, 699-674.
- El-sherbiny, M. M. , Alhamali, R. M, (2011), A Hybrid Particle Swarm Method with Artificial Immune Learning for Solving the Fixed Charge Transportation Problem, submitted to *Computers & Industrial Engineering*.
- Gen, M., K. Ida, and Y.Z. Li, (1998). Bicriteria transportation problem by hybrid genetic algorithm. *Computers and Industrial Engineering*, 35(12) 363-366.
- Gottlieb, J., B.A.Julstrom, F.Rothlauf, and Raidl, (2001). G.R.: Prüfer numbers: A poor representation of spanning trees for evolutionary search. In: Spector L, et al., eds. *Proc. of the 2001 Genetic and Evolutionary Computation Conf.* San Francisco: Morgan Kaufmann Publishers, 343-350.
- Hajiaghaei, M. Keshteli, S. Molla-Alizadeh-Zavardehi, R. Tavakkoli-Moghaddam, (2010). Addressing a nonlinear fixed-charge transportation problem using a spanning tree-based genetic algorithm, *Computers & Industrial Engineering*, 59, 259-271.
- Kannan, G., Sasi Kumar, P., & Vinay, V. P. (2008) Comments on "Nonlinear fixed charge transportation problem by spanning tree-based genetic algorithm" by Jung-Bok Jo, Yinzhen Li, Mitsuo Gen, *Computers & Industrial Engineering* (2007). *Computers & Industrial Engineering*, 55, 533-534.
- McKeown, P. G., (1975). A vertex ranking procedure for solving the linear fixed charge problem. *Operations Research*, 1183-1191.
- Murty, K.G., (1968). Solving the fixed charge problem by ranking the extreme points, *Operations Research*, 268-279
- Orhan Engin, Alper Döyen, (2004). A new approach to solve hybrid flow shop scheduling problems by artificial immune system, *Future Generation Computer Systems*, 20, 1083-1095.
- Palekar, U.S., M.K. Karwan, and S.Zionts, (1990). A branch-and bound method for the fixed charge transportation problem, *Management Science*, 36(9): 1092-1105.

partially loaded. As in step 3, the shipped quantities through these routes can be perturbed to obtain the desirable improved distribution as shown in Table 11.

Table 11: Improved distribution for the second problem after first iteration

	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	D ₈
S ₁		100				50	10	
S ₂				100		50		
S ₃	20		70				80	
S ₄				140				
S ₅	120							40
S ₆			100					
S ₇						50	70	
S ₈					60			140

The corresponding variable cost will be **12680**, the opening cost will be **36500**, and the total cost will be **49180**.

In the second iteration, it can be observed that using step 2, the shipped quantities through the routes (3, 1), (3,3), (5,1) and (5,3) can be perturbed to obtain the desirable improved distribution as shown in Table 12.

Table 12: Improved distribution for the second problem after second iteration

	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	D ₈
S ₁		100				50	10	
S ₂				100		50		
S ₃	40		50				80	
S ₄				140				
S ₅	100		20					40
S ₆			100					
S ₇						50	70	
S ₈					60			140

The variable cost = **12780**, the opening cost = **35500**, and the total cost is **48280**

In the third iteration, it can be observed that using step 2, the shipped quantities through the routes (1, 1), (1,7), (3,1) and (3, 7) can be perturbed to obtain the desirable improved distribution as shown in Table 13.

Table 13: Improved distribution for the second problem after third iteration

	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	D ₈
S ₁	10	100				50		
S ₂				100		50		
S ₃	30		50				90	
S ₄				140				
S ₅	100		20					40
S ₆			100					
S ₇						50	70	
S ₈					60			140

The variable cost = **12820**, the opening cost = **35100**, and the total cost is **47920**

Table 8: The coefficient matrix for the first problem

	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	D ₈
S ₁	26.43	34.00	57.00	59.00	64.00	37.00	42.00	58.00
S ₂	33.71	37.00	45.00	47.00	69.00	32.00	37.00	47.00
S ₃	26.14	37.00	50.29	58.71	62.67	40.00	40.00	48.29
S ₄	39.14	44.00	50.29	45.14	58.00	40.14	36.71	45.57
S ₅	25.29	45.00	52.00	56.00	62.33	41.00	43.00	46.50
S ₆	29.00	48.00	40.00	48.00	65.00	35.00	35.00	45.00
S ₇	32.50	38.00	63.00	62.50	73.67	37.00	38.00	55.00
S ₈	33.71	52.00	48.24	55.00	31.67	40.00	44.50	43.56

Table 9: A local optimal distribution for the second problem based on coefficient matrix in Table 8

	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	D ₈
S ₁		100				60		
S ₂				100		50		
S ₃	20		70				80	
S ₄				140				
S ₅	120							40
S ₆			100					
S ₇						40	80	
S ₈					60			140

Number of trucks corresponding to the local optimal distribution presented in Table 9 is presented in Table 10.

Table 10: Number of trucks corresponding to Table 9

	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	D ₈
S ₁		2				3		
S ₂				3		3		
S ₃	3		4				4	
S ₄				3				
S ₅	3							4
S ₆			2					
S ₇						3	3	
S ₈					2			4

The shipping cost for the local optimal distribution for the first problem can be calculated as follows:

$c_{12}x_{12} + c_{16}x_{16} + c_{24}x_{24} + c_{26}x_{26} + c_{31}x_{31} + c_{33}x_{33} + c_{37}x_{37} + c_{44}x_{44} + c_{51}x_{51} + c_{58}x_{58} + c_{63}x_{63} + c_{76}x_{76} + c_{77}x_{77} + c_{85}x_{85} + c_{88}x_{88} = 10 \times 100 + 9 \times 60 + 11 \times 100 + 8 \times 50 + 9 \times 20 + 15 \times 70 + 15 \times 80 + 13 \times 140 + 6 \times 120 + 9 \times 40 + 18 \times 100 + 7 \times 40 + 8 \times 80 + 5 \times 60 + 8 \times 140 = 12710$. Also, the total opening costs based on Table 10 can be calculated as follows: $t_{12}f_{12} + t_{16}f_{16} + t_{24}f_{24} + t_{26}f_{26} + t_{31}f_{31} + t_{33}f_{33} + t_{37}f_{37} + t_{44}f_{44} + t_{51}f_{51} + t_{58}f_{58} + t_{63}f_{63} + t_{76}f_{76} + t_{77}f_{77} + t_{85}f_{85} + t_{88}f_{88} = 2 \times 1200 + 3 \times 1400 + 3 \times 1800 + 3 \times 1200 + 3 \times 800 + 4 \times 1500 + 4 \times 1000 + 3 \times 1500 + 3 \times 900 + 4 \times 1500 + 2 \times 1000 + 3 \times 1200 + 3 \times 1200 + 2 \times 800 + 4 \times 1600 = 36500$. Therefore, the total cost is 49210.

As the first iteration, it can be observed that using step 2, there is a rectangular closed path through the routes (1,6), (1,7), (7,6) and (7,7) where the diagonally opposite cells (1,6) and (7,7) are not multiples of A_{ij} and hence

The shipping cost for the local optimal distribution for the first problem can be calculated as follows:

$c_{13}x_{13} + c_{15}x_{15} + c_{25}x_{25} + c_{32}x_{32} + c_{34}x_{34} + c_{35}x_{35} + c_{41}x_{41} + c_{45}x_{45} = 9 \times 50 + 1 \times 10 + 3 \times 10 + 7 \times 20 + 3 \times 10 + 5 \times 10 + 8 \times 40 + 8 \times 50 = 1430$. Also, the total opening costs based on Table 4 can be calculated as follows: $t_{13}f_{13} + t_{15}f_{15} + t_{25}f_{25} + t_{32}f_{32} + t_{34}f_{34} + t_{35}f_{35} + t_{41}f_{41} + t_{45}f_{45} = 2 \times 100 + 1 \times 500 + 1 \times 200 + 1 \times 100 + 1 \times 200 + 1 \times 300 + 2 \times 100 + 2 \times 100 = 1900$. Therefore, the total cost is 3330.

As the first iteration, it can be observed that using step 2, there is a rectangular closed path through the routes (1,1), (1,5), (4,1) and (4,5) where the diagonally opposite cells (1,5) and (4,1) are not multiples of A_{ij} and hence partially loaded. As in step 3, the shipped quantities through these routes can be perturbed to obtain the desirable improved distribution as shown in Table 5.

Table 5: Improved distribution for the first problem after first iteration

	D ₁	D ₂	D ₃	D ₄	D ₅
S ₁	10		50		
S ₂					10
S ₃		20		10	10
S ₄	30				60

The corresponding variable cost will be 1470, the opening cost will be 1700, and the total cost will be 3170.

In the second iteration, it can be observed that using step 2, the shipped quantities through the routes (2, 2), (2,5), (3,2) and (3, 5) can be perturbed to obtain the desirable improved distribution as shown in Table 6.

Table 6: Improved distribution for the first problem after second iteration

	D ₁	D ₂	D ₃	D ₄	D ₅
S ₁	10		50		
S ₂		10			
S ₃		10		10	20
S ₄	30				60

The variable cost = 1440, the opening cost = 1700, and the total cost is 3140.

It can be observed that Table 6 provides an improved solution to the problem which meets the demand and supply constraints.

4.2. The second example

The parameters and variables of the second problem are given in Table 7.

Table 7: The parameters and variables of example 4×5

	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
	140	100	170	240	60	150	160	180								
s_i	Variable cost c_{ij}								Route opening cost f_{ij}							
160	5	10	12	9	4	9	7	8	1000	1200	1800	2000	1800	1400	1400	2000
150	8	15	11	11	9	8	5	11	1200	1100	1700	1800	1800	1200	1600	1800
170	9	17	15	14	6	8	15	13	800	1000	1500	1900	1700	1600	1000	1500
140	7	14	16	13	8	8	11	7	1500	1500	1600	1500	1500	1500	1200	1800
160	6	17	17	16	9	5	8	9	900	1400	1400	1600	1600	1800	1400	1500
100	5	12	20	18	5	9	9	5	1200	1800	1000	1500	1800	1300	1300	2000
120	5	14	18	20	7	7	8	5	1100	1200	1800	1700	2000	1200	1200	2000
200	8	18	20	15	5	8	7	8	1200	1700	1200	2000	800	1600	1500	1600

Step value $A_{ij} = 50 \forall i, j$.

Step 3: Perturb the units in the cells of the rectangular closed path in such a way that supply and demand constraints are not violated.

Step 4: Determine desirability of the move.

Step 5: Repeat Steps 2 through 4 until all empty cells have been evaluated.

4. Illustrative Examples

This section represents two illustrative examples used to demonstrate the application of the proposed algorithm for solving FMSFCTP. The examples considered are of dimension 4x5 and 8x8 with variable opening costs. In these examples the step values are considered as a constant value for all routes.

4.1.The first example

The parameters and variables of the first problem are given in Table 1.

Table 1: The parameters and variables of example 4x5

	1	2	3	4	5	1	2	3	4	5
	d_i									
	40	20	50	10	80					
s_i	Variable cost c_{ij}					Route opening cost f_{ij}				
60	5	4	9	2	1	400	300	100	400	500
10	5	2	9	9	3	300	200	200	300	200
40	5	7	2	3	5	400	100	300	200	300
90	8	1	8	1	8	100	800	400	500	100

Step value $A_{ij} = 30 \forall i, j$.

Table 2: The coefficient matrix for the first problem

	D ₁	D ₂	D ₃	D ₄	D ₅
S1	25.00	19.00	13.00	42.00	17.67
S2	35.00	22.00	29.00	39.00	23.00
S3	25.00	12.00	17.00	23.00	20.00
S4	13.00	41.00	24.00	51.00	11.75

Table 3: A local optimal distribution for the first problem based on coefficient matrix in Table 2

	D ₁	D ₂	D ₃	D ₄	D ₅
S ₁			50		10
S ₂					10
S ₃		20		10	10
S ₄	40				50

Number of trucks corresponding to the local optimal distribution presented in Table 3 is presented in Table 4.

Table 4: Number of trucks corresponding to Table 3

	D ₁	D ₂	D ₃	D ₄	D ₅
S ₁			2		1
S ₂					1
S ₃		1		1	1
S ₄	2				2

A critical look at (17) reveals that the formulation fails to consider the cases when $A_{ij} = M_{ij}$ and $A_{ij} > M_{ij}$ as the values will be infinity when $A_{ij} = M_{ij}$ and assumes negative value in case $A_{ij} > M_{ij}$. Hence the proposed methodology concentrates on the concepts of generating the formulae (15) and (16).

A critical look at the formula (16) proposed by Kowalski and Lev, it can be observed that the expected cost per unit for each cell C_{ij} is split into two parts. The first being the expected fixed cost per unit and the second being the variable cost per unit.

A similar concept is applied in developing a methodology for solving the proposed model for the MFSFCTP presented in (12) to (14), the expected number of units transported through the route (i,j) is M_{ij} . The number of trucks needed to transport these M_{ij} units through the route (i,j) will be $\left\lceil \frac{M_{ij}}{A_{ij}} \right\rceil$ and the expected total opening cost will be $\left\lceil \frac{M_{ij}}{A_{ij}} \right\rceil f_{ij}$. Further, the expected fixed cost per unit, which is referred to as the expected opening cost per unit, is determined by $\left\lceil \frac{M_{ij}}{A_{ij}} \right\rceil f_{ij} / M_{ij}$. Thus the expected cost per unit for transporting M_{ij} units through the route (i,j) can be determined as in (18)

$$C_{ij} = \left\lceil \frac{M_{ij}}{A_{ij}} \right\rceil f_{ij} / M_{ij} + c_{ij} \quad \forall i, j \quad (18)$$

In case of $M_{ij} < A_{ij}$, the number of trucks needed for transportation will be one truck. Hence the opening cost in such case will be equal to f_{ij} which is same as the case denoted by (15). Thus the proposed formula (18) imbibes the concepts of formulating the formulae (15) and (16) for constructing the coefficient matrix as an intermediate stage for solving the SFCTP.

The basic idea of the proposed algorithm is constructing the coefficient matrix using (18), solving the resulting traditional transportation problem, and improving the resulting solution using the stepping stone method. The steps in the pseudo code of the algorithm are as follows:

- Step 1: Find an initial solution by translating the problem to the traditional transportation problem based on the coefficient matrix computed using (18)
- Step 2: Pick any empty cell and identify the rectangular closed path leading to that cell such that the diagonally opposite assigned cells are not multiples of A_{ij} and hence partially loaded. In the rectangular closed path there can only be one empty cell that is being examined. The 90-degree turns must therefore occur at those places that meet this requirement.

Where $\lceil x \rceil$ is the ceiling function that return the integer number greater than or equal to x .

3. The methodology of solving FMSFCTP

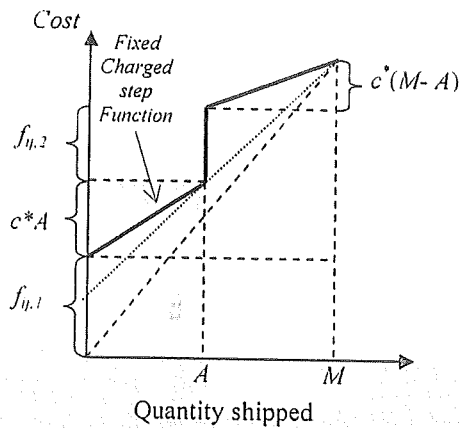


Fig. 1. Shipping costs as function of quantity shipped along route (i, j) for SFCTP

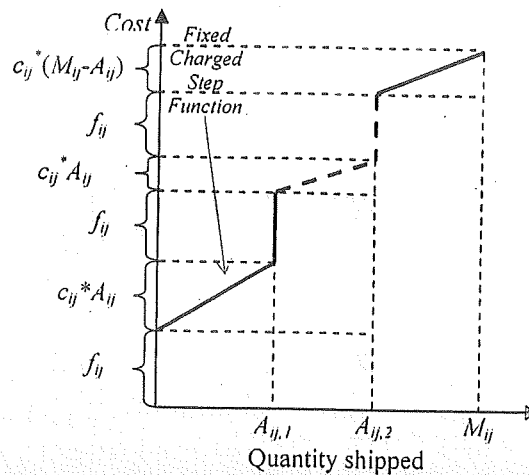


Fig. 2. Shipping costs as function of quantity shipped along the route (i, j) for FMSFCTP

As illustrated in Fig. 1, in the case of SFCTP, for every loaded route (i, j) the cost of the fixed-charge step function formulation is greater than the corresponding cost of the relaxed integer restriction function. The situation in case of FMSFCTP is illustrated in Fig. 2.

As stated earlier, not much work has been done concerning solution of SFCTPs. The existing methods for dealing with SFCTPs are based on using a certain formula for converting the problem into a classical transportation problem by constructing an intermediate coefficient matrix using a formula and finding the solution thereafter.

(Balinski, 1961) has provided a heuristic solution for FCTP by considering the unit transportation cost of shipping through the route (i, j) by using the formula as in (15) for constructing the intermediate coefficient matrix mentioned above.

$$C_{ij} = f_{ij,1} / M_{ij} + c_{ij} \quad (15)$$

where $M_{ij} = \min(S_i, D_j)$

In order to improve the local solution of the classical transportation problem found from converting SFCTP Kowalski and Lev (2008) have proposed a heuristic technique for improving such solution. They have proposed two heuristic algorithms. In the first algorithm, the formula considered was as in (16) and in the second algorithm, the formula considered was as in (17).

$$C_{ij} = (f_{ij,1} + f_{ij,2}) / M_{ij} + c_{ij} \quad (16)$$

$$C_{ij} = f_{ij,2} / (M_{ij} - A_{ij}) + c_{ij} \quad (17)$$

$$\text{Min } Z = \sum_{i=1}^m \sum_{j=1}^n (c_{ij}x_{ij} + \sum_{l=1}^{k+1} y_{ij,l}f_{ij,l}) \quad (9)$$

$$\text{s.t. } \sum_{i=1}^m x_{ij} = d_j \quad \text{for } j = 1, \dots, n \quad (10)$$

$$\sum_{j=1}^n x_{ij} = s_i \quad \text{for } i = 1, \dots, m \quad (11)$$

$$x_{ij} \geq 0 \quad \forall i, j$$

$$y_{ij,l} = \begin{cases} 1 & \text{if } x_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j, l$$

$$\sum_{i=1}^m s_i = \sum_{j=1}^n d_j$$

2.4. Formulation of MFSFCTP

In the proposed Multi Flat Step Fixed Charge Transportation Problem, the problem considered is a new version of SFCTP. The proposed version considers the problem of minimizing transportation cost in a situation where the items are transported through trucks or wagons and the capacity of the truck determines the number of trucks engaged for transportation. The corresponding cost f_{ij} for route (i,j) is assumed to be the opening cost for the route and remains flat up to a certain quantity A_{ij} to be shipped which corresponds to the capacity of the truck and increases in multiples of this flat cost f_{ij} for each additional truck employed for transporting through this route, in steps of A_{ij} . The number of trucks engaged will be $\left\lceil \frac{x_{ij}}{A_{ij}} \right\rceil$ where $\lceil \cdot \rceil$ is the ceiling function. The shipping cost through the route (i,j) can be represented as the sum of the variable cost $c_{ij}x_{ij}$ and the step opening cost denoted by $\left\lceil \frac{x_{ij}}{A_{ij}} \right\rceil f_{ij}y_{ij}$, where y_{ij} is a binary number which represents whether the route (i,j) is used or not as represented in (4). The mathematical model of the proposed MFSFCTP can be represented as follows:

$$\text{Min } Z = \sum_{i=1}^m \sum_{j=1}^n (c_{ij}x_{ij} + \left\lceil \frac{x_{ij}}{A_{ij}} \right\rceil f_{ij}y_{ij}) \quad (12)$$

$$\text{s.t. } \sum_{i=1}^m x_{ij} = d_j \quad \text{for } j = 1, \dots, n \quad (13)$$

$$\sum_{j=1}^n x_{ij} = s_i \quad \text{for } i = 1, \dots, m \quad (14)$$

$$x_{ij} \geq 0 \quad \forall i, j$$

$$y_{ij} = \begin{cases} 1 & \text{if } x_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j$$

$$\sum_{i=1}^m s_i = \sum_{j=1}^n d_j$$

$$\text{Min } Z = \sum_{i=1}^m \sum_{j=1}^n (c_{ij} x_{ij} + b_{ij,1} f_{ij,1} + b_{ij,2} f_{ij,2}) \quad (5)$$

$$\text{s.t. } \sum_{i=1}^m x_{ij} = d_j \quad \text{for } j = 1, \dots, n \quad (6)$$

$$\sum_{j=1}^n x_{ij} = s_i \quad \text{for } i = 1, \dots, m \quad (7)$$

$$x_{ij} \geq 0 \quad \forall i, j$$

$$b_{ij,1} = \begin{cases} 1 & \text{if } x_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j$$

$$b_{ij,2} = \begin{cases} 1 & \text{if } x_{ij} > A_{ij,1} \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j$$

$$\sum_{i=1}^m s_i = \sum_{j=1}^n d_j$$

2.3. Formulation of MSFCTP

In Multi Step Fixed Charge Transportation Problem (MSFCTP) the fixed cost f_{ij} for route (i, j) is proportional to the transported amount through this route. So, an additional cost is added when the transported units exceeds certain amounts $A_{ij,1}, A_{ij,2}, A_{ij,3}, \dots, A_{ij,k}$. The fixed cost f_{ij} can be illustrated as follows:

Assume that $f_{ij,1}$ is the fixed cost to open a route (i, j) if the transported quantity x_{ij} is less than or equal to a certain amount $A_{ij,1}$. When the quantity exceeds this amount $A_{ij,1}$, an additional fixed cost is added to the same route which is $f_{ij,2}$, if the quantity is less than or equal to a certain amount $A_{ij,2}$. Similarly, when the transported quantity x_{ij} exceeds this amount $A_{ij,2}$, an additional fixed cost is added to the same route which is $f_{ij,3}$ until the quantity transported is $A_{ij,3}$ and so on. In general, we can consider k steps in the fixed cost. Therefore the fixed cost f_{ij} can be calculated by (4).

$$f_{ij} = b_{ij,1} f_{ij,1} + b_{ij,2} f_{ij,2} + \dots + b_{ij,k+1} f_{ij,k+1}$$

$$f_{ij} = \sum_{l=1}^{k+1} b_{ij,l} f_{ij,l} \quad \forall i, j \quad (8)$$

where

$$b_{ij,1} = \begin{cases} 1 & \text{if } x_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j$$

$$b_{ij,2} = \begin{cases} 1 & \text{if } x_{ij} > A_{ij,1} \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j$$

$$\vdots \quad \vdots \quad \vdots$$

$$b_{ij,k+1} = \begin{cases} 1 & \text{if } x_{ij} > A_{ij,k} \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j$$

$$A_{ij,2} > A_{ij,1} > \dots > A_{ij,k} \quad \forall i, j$$

Combining Eq. (8) in the FCTP mathematical model the standard mathematical model of the Multi-Step Fixed Charge Transportation Problem (MSFCTP) can be represented as follows:

2. Description and formulation

This section presents the steps in developing the MSFCTP and its special version, starting from the Fixed Charge Transportation Problem (FCTP). The mathematical models of the problems are also developed and presented.

2.1. Fixed Charge Transportation Problem

FCTP can be described as a distribution problem in which there are m suppliers (warehouses or plants or factories) and n customers (destinations or demand points). Each of the m suppliers can ship to any of the n customers at a shipping cost per unit c_{ij} (unit cost for shipping from supplier i to customer j) plus a fixed cost f_{ij} , assumed for opening this route. Each supplier $i=1, 2, \dots, m$ has s_i units of supply and each customer $j=1, 2, \dots, n$ demands d_j units. x_{ij} is the unknown quantity to be transported on the route (i, j) from plant i to customer j . The objective is to determine which routes are to be opened and the size of the shipment, so that the total cost of meeting demand, given the supply constraints, is minimized. The mathematical model of FCTP can be represented as follows:

$$\text{Min } z = \sum_{i=1}^m \sum_{j=1}^n (c_{ij}x_{ij} + f_{ij}y_{ij}) \quad (1)$$

$$\text{s.t. } \sum_{i=1}^m x_{ij} = d_j \quad \text{for } j = 1, \dots, n \quad (2)$$

$$\sum_{j=1}^n x_{ij} = s_i \quad \text{for } i = 1, \dots, m \quad (3)$$

$$x_{ij} \geq 0 \quad \forall i, j$$

$$y_{ij} = \begin{cases} 1 & \text{if } x_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j \quad (4)$$

$$x_{ij} \geq 0 \quad \text{and} \quad \sum_{i=1}^m s_i = \sum_{j=1}^n d_j$$

2.2. Step Fixed Charge Transportation Problem

In Step Fixed Charge Transportation Problem (SFCTP) the cost of shipping through route (i, j) consists of a variable cost equal to $x_{ij}c_{ij}$ plus a fixed cost f_{ij} . The fixed cost f_{ij} for route (i, j) is proportional to the transported amount through its route. This consists of a fixed cost $f_{ij,1}$ for opening the route (i, j) and an additional cost $f_{ij,2}$ when the transported units exceeds a certain amount A_{ij} . The objective is to determine which routes are to be opened and the size of the shipment, so that the total cost of meeting demand, given the supply constraints, is minimized. The standard mathematical model of Step Fixed Charge Transportation Problem (SFCTP) can be represented as follows (Altashan et al., 2012):

method (McKeown, 1975), and the branch-and-bound method (Palekar et al., 1990) amongst others. Exact methods are, however, generally not very useful when a problem reaches a certain level, because they do not make the most use of the special network structure of the FCTP.

Therefore, heuristic methods have been proposed, such as the adjacent extreme point search method (Balinski, 1961; Sun et al., 1993), and the Lagrangian relaxation method (Wright et al., 1989, 1991) and such other heuristic methods (Kowalski and Lev, 2007, 2008; Adlakha et al., 2007; Adlakha et al., 2010; Lev and Kowalski, 2011). Although these methods are usually computationally efficient, the major disadvantage of heuristic methods is the possibility of terminating at a local optimum that is far distant from the global optimum. Recently, some meta-heuristic methods employed in the FCTP, such as the Tabu search method for FCTPs (Sun et al., 1998), genetic algorithms (GA) based on a spanning tree with Prüfer numbers (Gen et al., 1998), and genetic algorithms based on a matrix permutation representation (Gottlieb et al., 2001), (Raidl and Julstrom, 2003), have improved the effective coding of the spanning tree method based on edge sets.

One of the versions of FCTP is the Step Fixed-charge transportation problem (SFCTP) where the fixed cost is incurred for every route that is used in the solution. In SFCTP, the fixed cost is proportional to the amount shipped. The proposed Flat Multi Step Fixed Charge Transportation Problem (FMSFCTP) is a new version of SFCTP which considers the problem of minimizing transportation cost in a situation where an opening cost is incurred for every route used in the solution and remains flat up to a certain quantity and increases in multiples of this flat cost. This cost structure invokes the ceiling function and causes the value of the objective function to behave like a step function. Since problems with fixed charge are usually NP-hard (Nondeterministic Polynomial-time), the computational time to obtain exact solutions increases in a polynomial fashion and very quickly becomes extremely difficult and long as the size of the problem increases. In the case of the SFCTP due to the step function structure of the objective function, we are dealing with a "NP-super hard" problem with much "higher degree" of the polynomial complexity (Kowalski and Lev, 2008). Since FMSFCTP is an extended version of SFCTP, it is also a "NP-super hard" problem. Two problems with different dimensions have been solved to evaluate and demonstrate the performance of the proposed algorithm.

The rest of the paper is organized as follows: in section 2, MSFCTP and one of its special versions are described with their mathematical formulations. In section 3, the proposed heuristic algorithm is presented.

Numerical illustrations with proposed algorithm are presented in section 4. Finally, the conclusion and future work are reported in section 5.

On Multi-Step Fixed Charge Transportation Problem

Ibrahim A. Aljasser¹

Mahmoud M. El-Sherbiny²

Bokkasam Sasidhar¹

Abstract

Fixed charge transportation problem (FCTP) is considered to be an NP-hard problem. One of its versions is the Step Fixed charge transportation problem (SFCTP) where the fixed cost is incurred for every route that is used in the solution. The general form of Multi Step Fixed Charge Transportation Problem (MSFCTP) and one of its special versions – Multi Flat Step Fixed Charge Transportation Problem (MFSFCTP) are proposed as new versions of SFCTP. In MFSFCTP the problem of minimizing transportation cost is considered in situations where an opening cost is incurred for every route used in the solution and increases in multiples of shipped quantity. The special version of MFSFCTP considers the opening cost to remain flat up to a certain quantity and increases in multiples of this flat cost. This cost structure invokes the ceiling function and causes the value of the objective function to behave like a step function. This paper describes and presents the mathematical models of MSFCTP and MFSFCTP together with a heuristic algorithm for solving MFSFCTP. Two problems have been solved to evaluate and demonstrate the performance of the proposed algorithm.

Keywords: *Fixed charge transportation, Step fixed charge transportation, Heuristic algorithm, Flat multi step fixed charge transportation.*

1. Introduction

The fixed-charge problem was first formulated in 1954 (Hirsch and Dantzig, 1954). During 1961 (Balinski,1961) it was shown that fixed-charge transportation problem is a special case of fixed-charge problem and an approximate solution was presented. Since then, considerable research has been carried out on this topic. In 1988 Sandrock analyzed the source induced fixed-charge transportation problem (Sandrock,1988). Generally, solving methods can be classified as exact or heuristic methods. Exact methods for solving the FCTP include the cutting planes method (Rousseau, 1973), the vertex ranking

¹ Quantitative Analysis Dept., College of Business Administration, King Saud University, Saudi Arabia, ialjasser@ksu.edu.sa,

² Operations Research Dept., Institute of Statistical Studies & Research (ISSR), Cairo University, Egypt. m_sherbiny@yahoo.com,

³ Quantitative Analysis Dept., College of Business Administration, King Saud University, Saudi Arabia. bbokkasam@ksu.edu.sa

6	3	31	14.462	14.545	16	0
6	3	8	16.805	16.888	0	5
6	4	37	19.005	19.088	15	0
6	4	38	19.146	19.23	13	0
6	4	41	19.413	19.496	27	0
6	4	41	19.496	19.58	0	76
6	4	38	19.763	19.846	0	36
6	4	37	19.905	19.988	0	8
6	5	31	27.621	27.705	28	0
6	5	33	27.788	27.871	22	0
6	5	31	27.955	28.038	0	56
6	5	28	28.255	28.338	0	9
6	5	25	28.471	28.555	0	24

3	4	31	15.324	15.408	12	0
3	4	33	15.488	15.571	19	0
3	5	4	20.255	20.338	8	0
3	5	4	20.338	20.421	0	120
3	6	42	25.121	25.205	17	0
3	6	45	25.355	25.438	4	0
3	6	45	25.438	25.521	0	22
3	6	42	25.671	25.755	0	98
4	1	53	8.09	8.173	10	0
4	1	55	8.379	8.462	20	0
4	1	42	9.694	9.777	90	0
4	1	42	9.777	9.86	0	33
4	2	10	14.22	14.304	18	0
4	2	54	14.804	14.887	7	0
4	2	54	14.937	15.02	0	94
4	2	10	15.713	15.796	0	10
4	2	8	15.938	16.021	0	16
4	3	25	18.538	18.621	79	0
4	3	27	18.68	18.763	41	0
4	3	27	18.763	18.846	0	68
4	3	25	18.905	18.988	0	52
4	4	31	24.33	24.413	12	0
4	4	32	24.43	24.513	15	0
4	4	33	24.58	24.663	93	0
4	4	33	24.663	24.746	0	14
4	4	32	24.813	24.896	0	52
4	4	31	24.913	24.996	0	54
5	1	45	9.079	9.162	18	0
5	1	41	9.37	9.453	68	0
5	1	41	9.453	9.536	0	34
5	1	22	10.237	10.32	0	45
5	2	16	14.179	14.262	1	0
5	2	10	14.795	14.879	0	120
5	3	6	17.421	17.505	0	106
5	4	4	21.113	21.196	12	0
5	4	4	21.196	21.28	0	12
5	5	42	28.413	28.496	25	0
5	5	42	28.763	28.846	0	17
6	1	4	8.199	8.282	17	0
6	1	4	8.282	8.366	0	10
6	2	4	9.378	9.461	10	0
6	2	4	9.461	9.545	0	15
6	3	22	12.07	12.154	29	0
6	3	37	12.462	12.545	29	0
6	3	38	12.604	12.687	23	0
6	3	41	12.87	12.954	7	0
6	3	42	13.012	13.095	16	0

Appendix

Table 6: Timetable of coal mining rail transport system

Result						
Train index	Run Index	Mine Index	Start Time	Finish Time	Empty wagons delivered	Full wagons collected
1	1	54	7.75	7.833	54	0
1	1	55	7.858	7.945	66	0
1	1	55	7.945	8.029	0	60
1	2	31	10.762	10.845	82	0
1	2	32	10.862	10.945	37	0
1	2	31	10.962	11.045	0	40
1	2	28	11.262	11.345	0	6
1	2	27	11.42	11.504	0	42
1	2	25	11.562	11.645	0	32
1	3	22	15.563	15.646	67	0
1	3	22	15.73	15.813	0	51
1	3	8	16.446	16.53	0	69
1	4	6	17.946	18.03	13	0
1	4	6	18.03	18.113	0	6
1	5	41	22.321	22.405	8	0
1	5	37	22.646	22.73	0	36
2	1	4	7.35	7.433	120	0
2	1	4	7.433	7.517	0	17
2	2	8	8.442	8.529	58	0
2	2	53	9.444	9.528	40	0
2	2	53	9.528	9.611	0	50
2	3	16	12.295	12.379	57	0
2	3	16	12.379	12.462	0	98
2	4	10	15.455	15.538	52	0
2	4	54	16.08	16.163	33	0
2	4	55	16.188	16.271	35	0
2	4	55	16.271	16.355	0	116
2	5	55	19.146	19.23	55	0
2	5	27	20.23	20.313	44	0
2	5	28	20.388	20.471	12	0
2	5	33	20.771	20.855	0	120
3	1	6	7.225	7.308	99	0
3	2	10	8.191	8.274	60	0
3	2	16	8.545	8.629	40	0
3	3	4	10.237	10.32	7	0
3	4	8	13.362	13.445	32	0
3	4	25	14.679	14.762	29	0
3	4	27	14.82	14.904	25	0
3	4	28	14.979	15.062	3	0

applying the algorithms. A mining railway operation is very complex and has a huge number of variables. In the proposed model, the system constraints can be classified into two main categories. Firstly, there are constraints related to the rail operations system. These constraints are very important to ensure passing of trains without accidents or delays. Secondly, there are constraints related to mining rail operations such as the train, siding and port capacity, empty wagon requirements, and full wagon requirements. The main advantage of this model is reactivity. Requirements for wagons are achieved without interruption to their operations. The main outputs of the model are efficient schedules for coal mining transport systems to optimise the performance of the system. These schedules considered train passing constraints to satisfy safety conditions in the system.

REFERENCES

- Burdett, R.L., Kozan, E. (2014) Determining Operations Affected by Delay in Predictive Train Timetables. *Computers & OR* 41: 150-166.
- Burdett, R.L., Kozan, E. (2009) A sequencing Approach for Creating New Train Timetables. *OR Spectrum*, DOI 10.1007/s00291-008-0143-6
- Burdett, R.L., Kozan, E. (2010) A disjunctive Graph Model and Framework for Constructing New Train Schedules. *European Journal of Operational Research* 200:85-98.
- Kozan, E., Liu, S.Q. (2012): A demand-Responsive Decision Support System for Coal Transportation. *Decision Support Systems* 54(1): 665-680.
- Henz M, Müller T, Thiel S. (2004) Global Constraints for Round-Robin Tournament Scheduling. *European Journal of Operational Research*, 153(1):92-101.
- Her, J.H., Ramakrishna, R.S. (2007) An external-Memory Depth-First Search Algorithm for General Grid Graphs. *Theoretical Computer Science*, 374: 170-180.
- Janiak, A., Kovalyov, Y., Kozan, E., Pesch, E. (2013): Parallel machine Scheduling and Common Due Window Assignment with Job Independent Earliness and Tardiness Costs. *Inf. Sci.* 224: 109-117.
- Masoud M, Kozan E, Kent G (2010) Scheduling Techniques to Optimise Sugarcane Rail Systems. *ASOR Bulletin* 29:25-34
- Liu, S. Q., Kozan, E., 2009. Scheduling Trains as a Blocking Parallel-Machine Job Shop Scheduling Problem. *Computer and Operations Research*, 36(10): 2840-2852.
- Russell, R. A., Urban, T. L. (2006) A constraint Programming Approach to The multiple-Venue, Sport-Scheduling Problem. *Computers & Operations Research* 33: 1895-1906.
- Sierra, M. R., Varela, R. (2008) Pruning by Dominance in Best-First Search for The job Shop Scheduling Problem with Total Flow Time. *J Intell Manuf.* DOI 10.1007/s10845-008-0167-4.

Table 5: Collecting and delivering wagons, and visit times

Result						
Train index	Run index	Mine index	Start Time	Finish Time	Empty wagons delivered	Full wagons collected
1	1	S _{16,3}	7.758	7.842	48	0
1	1	S _{25,3}	9.992	10.075	71	0
1	1	S _{23,3}	10.725	10.808	0	40
1	1	S _{16,3}	12.767	12.85	0	10
1	1	S _{9,3}	13.25	13.333	0	69
1	2	S _{23,3}	28.758	28.842	33	0
2	1	S _{18,3}	9.9	9.983	8	0
2	1	S _{16,3}	11.742	11.825	0	120
2	2	S _{23,3}	22.075	22.158	55	0
2	2	S _{27,3}	23.058	23.142	65	0
2	2	S _{27,3}	23.142	23.225	0	2
2	2	S _{25,3}	23.433	23.517	0	96
2	2	S _{23,3}	24.167	24.25	0	20
3	1	S _{27,3}	13.2	13.283	41	0
3	1	S _{23,3}	15.825	15.908	0	36
4	1	S _{9,3}	10.617	10.7	20	0
4	1	S _{18,3}	12.142	12.225	57	0
4	1	S _{18,3}	12.225	12.308	0	98
4	1	S _{9,3}	14.192	14.275	0	21
5	1	S _{9,3}	14.983	15.067	70	0
5	1	S _{16,3}	15.467	15.55	50	0
6	1	S _{16,3}	17.058	17.142	32	0
6	1	S _{18,3}	18.142	18.225	33	0
6	1	S _{23,3}	20.125	20.208	8	0
6	1	S _{25,3}	20.858	20.942	37	0
6	1	S _{27,3}	21.15	21.233	4	0
6	1	S _{27,3}	21.233	21.317	0	108
6	1	S _{25,3}	21.525	21.608	0	12

CONCLUSION

The proposed scheduling model is very complicated and needs to be solved in a reasonable time, because of the dynamic nature of the system and large number of variables. The CP approach is used to solve the coal rail transport problem. CP works well with all five search techniques under the two search strategies, except BFS with SSS. In addition, the DSS positively affected the CPU time for all search techniques after and before

the optimal solution by using DFS with SSS. Table 3 shows all solutions and CPU time to obtain each solution.

Table 3: Solution analysis under DFS and SSS

Solutions	Nb. of variables	Nb. of constraints	Nb. of choice points	Nb. of failure points	Solving time (s)	Makespan (hours)
First solution	10063	201310	3683	0	20.6	45.45
Second solution	10063	201310	8970	554	99.8	39.66
Third solution	10063	201310	11234	1435	189.5	35.5
Fourth solution	10063	201310	15670	4565	260	32.1
Fifth solution	10063	201310	18905	9812	434.2	29.1
Final solution	10063	201310	21208	11800	600.55	29.1

Run time is the time between starting the run at the port and returning to the port after visiting the sidings. Table 4 shows the processing run time of each run by each train.

Table 4: Runs processing time

Train index	Run index	Processing run time(hours)
1	1	6.67499971389771
1	2	4.33399963378906
2	2	7.99200057983398
2	1	4.50800037384033
3	1	9.29199981689453
4	1	4.35799980163574
5	1	1.66699886322021
6	1	8.52499961853027

Table 5 shows the start and finish times of each train run. All trains have to deliver and collect the siding allotment for the day (deliver empty wagons and collect full wagons). Table 5 shows the actual visit times for all trains at each siding, the sidings visited, the number of empty wagons delivered to and full wagons collected from each siding and run time. Run 2 was assigned to train 1 and the makespan satisfied at time 28.842.

In this paper a complete case study including 20 segment and 55 sections is solved see Appendix.

Table 2 shows the makespan and the CPU time for the two strategies with different search techniques. Also, as can be seen from this table, the number of variables and the constraints of the case study are very large. All these constraints need a lot of memory to be executed on a PC as a software code (see solver memory column in Table 2). The choice points column shows the explored points which can be branched and failure points which stop and cannot be extended in the search tree.

The Comparison between SSS and DSS in Table 2 clarifies that all search techniques have given results in a more reasonable and shorter time under DSS than when using SSS. DFS has achieved a good CPU time with the two strategies, while the BFS gave shorter times with DSS, whoever when combined with SSS, it was unable given any result (not applicable, n/a). As a result, we can consider the DFS technique is also suitable for solving the CP formulation.

The optimal solution for this case under the minimising the makespan objective was 28.842 hours at the mine area and 29.1 at port s. This solution was obtained with a CPU time 600.55s by using DFS under SSS and 89.12s using the DSS strategy, a reasonably short time. The result using different search techniques under SSS and DSS strategies are shown in Table 3.

Table 2: CPU time under minimising the makespan using DSS

Search technique	Nb. of variables	Nb. of constraints	Nb of choice points	Nb of failure points	SSS		DSS		Solver memory in bytes
					Solution (Makespan in hours)	CPU Time (s)	Solution (Makespan in hours)	CPU Time(s)	
DFS	10063	201310	21208	11800	29.1	600.55	29.1	89.12	323710724
SBS			21208	15006	29.1	950.12	29.1	140.15	410932124
DDS			23211	18980	29.1	810.50	29.1	121.88	558731012
BFS			16400	670	n/a	n/a	29.1	80.19	626613513
IDFS			37123	30150	29.1	689.34	29.1	109.66	410423468

Solution Analysis by Search Tree

After applying the CP search techniques for the model, in this section we try to explain how OPL has used the search tree techniques to obtain case study test solutions and how the search tree is built by branching techniques. For instance, we have obtained five solutions before obtaining

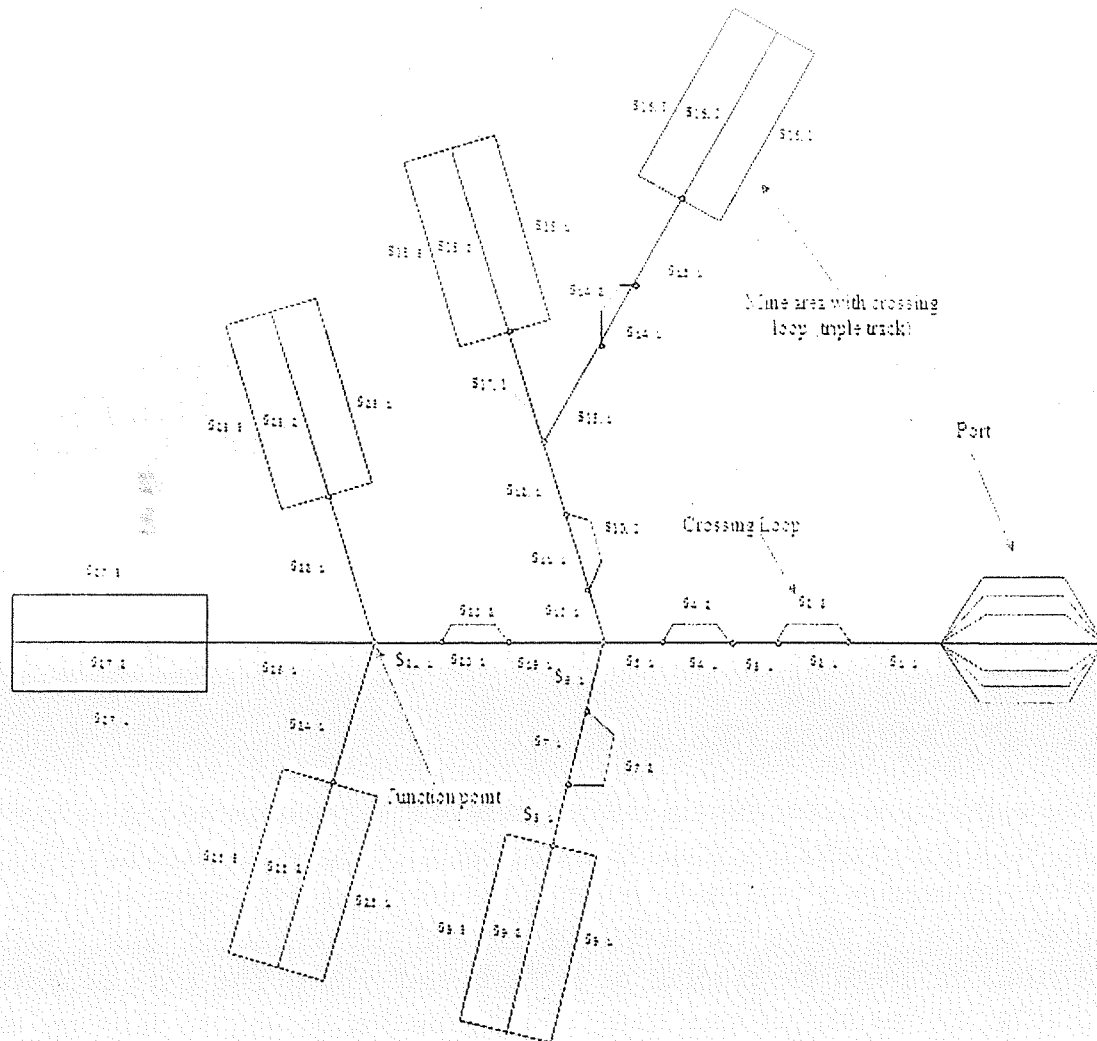


Figure 1: An artificial coal mining railway network

Table 1 shows the data of each train in the system such as the train speed, length, and capacity. 6 trains are included in the case study.

Table 1: Data of coal trains

Trains			
Train Index	Capacity Full wagons	Speed in km/h	Capacity Empty wagons
1	120	20	120
2	120	20	120
3	120	20	120
4	120	20	120
5	120	20	120
6	120	20	120

Interleaved depth first search (IDFS) works through DFS where IDFS prevents DFS falling into mistakes. While DFS searches sequentially from left to right, IDFS searches in parallel many subtrees at some tree level which is called active. It searches in each subtree to reach each leaf. If this leaf is the goal, the search process terminates. Otherwise, the state of the current subtree is recorded and added to a list of active subtrees. Slice Based Search (SBS) and Depth-bound Discrepancy Search (DDS) are effectives for job shop problems and will be used in this study.

Standard Strategy (SSS) and Dichotomic Search Strategy (DSS) will be integrated with the pervious search techniques inside OPL to improve the proformance of the previous search techniques. In SSS the objective function is evaluated at the first feasible solution point, and an upper bound is obtained and then the constraints are added to obtain the next solution. DSS depends on evaluating the objective function at the first feasiabile solution and at the mid value.

COMPUTATIONAL RESULTS

This paper uses a coal mining railway network to demonstrate and validate the developed models. A small sector of the rail transport system of mining areas was used as a case study to obtain the minimum Makespan of all train runs. Figure 1 shows the sector of the rail network used, along with information about the number of sections and mines. The trine capacity (number of wagons per each train) and the number of wagons allocated to each mine (allotments) will be shown in Table 1 and Table 2. The case study involves 15 single track sections, 6 double track, and 6 triple track trains (mine areas with crossing loops). There are 6 mines are for collecting, s_9 , s_{16} , s_{18} , s_{23} , s_{25} and s_{27} . Double track sections, s_2 , s_4 , s_7 , s_{10} , s_{16} , and s_{20} work as crossing loops without storage area, OPL software was used to obtain the solution. To satisfy both port and mines requirements from full and empty wagons, 12 runs are made by the 6 trains. The model is investigated using the Makespan as an objective function for this case study.

MODL SOLUTION TECHNIAUE

In Constraint programming approach, the global constraint technology is used to satisfy all the model constraints. Search techniques are used to solve the model constraints and investigate the model objective function value. The main advantage of the GCT is making sure all model constraints are satisfied before applying any solution techniques (search techniques) to investigate the feasibility of the model.

In this paper, the main solving technology of CP model in ILOG's OPL software is a global constraint satisfaction, which deals with problems defined within the finite set of possible values of each variable. This set is often called the domain of the variables. Most of all constraints of industrial applications use a finite domain. The constraint satisfaction problems are solved by using two main techniques: problem reduction and search techniques. Problem reduction is often referred to as consistency maintenance in the literature. This includes node consistency (NC), arc consistency (AC) and path consistency. Many efficient techniques have been developed to achieve AC and NC Constraints and are used for eliminating values from a variable's domain that do not satisfy the constraints, (Russell and Urban (2006); (Henz et al. 2004).

Many strategies are followed in ILOG's OPL software to find the solution through the search trees. The main advantage of these strategies is obtaining the optimal solution for many NP-hard problems but on the other side, using some of these techniques individually is time consuming and needs a huge memory to solve large scale problems. Depth-First Search Strategy (DFS) starts at the root node and proceeds by descending to its first descendant. Continuing in this process to reach a leaf. Then we backtrack to the parent of the leaf and descend to its next descendant, until it exists (Her and Ramakrishna 2007).

Best-first search (BFS) is a well-known strategy for optimization problems that consists of choosing the node with the best value of the objective function. For each job shop scheduling problem, we set the initial upper-bound on the Makespan to be the sum of the durations of all activities. If a solution is found with a lower Makespan, the new bound is added at the leaf node. Sierra and Varela (2008) used a Best First Search to reduce the search domain in solving the job shop scheduling problem with total flow time.

$$\alpha_{k_{\varepsilon} o'_{w'} s_{w'}} = 0 \quad (13)$$

End if

Equations (12- 13) ensure the collecting full wagons has to be in the inbound direction where o and o' are two operations at each siding in the two directions and o' precedes o.

Wagon Constraints

Equations (14) and (15) show the relation between the total allotments for each siding and the number of empty and full wagons delivered to or collected from each siding. These constraints will reduce the total number of wagons used in the transport process.

In equation (14), the total number of empty wagons delivered to siding s has to be equal to the allotment of this siding.

$$\sum_{k=1}^K \sum_{\varepsilon=1}^R \sum_{w=1}^h \sum_{o=1}^{O_w} \alpha_{k_{\varepsilon} o_{w'} s_{w'}} = A_{s_w} \quad \forall s. \quad (14)$$

In equation (15), the total number of full wagons collected from siding s has to be equal to the allotment of this siding.

$$\sum_{k=1}^K \sum_{\varepsilon=1}^R \sum_{w=1}^h \sum_{o=1}^{O_w} B_{k_{\varepsilon} o_{w'} s_{w'}} = A_{s_w} \quad \forall s. \quad (15)$$

Siding capacity:

Equation (16) ensures that the total number of empty and full wagons at each siding is less than the capacity of the siding.

$$\alpha_{k_{\varepsilon} o_{w'} s_{w'}} + B_{k_{\varepsilon} o_{w'} s_{w'}} \leq C_{s_w} \quad (16)$$

- In CP we cannot use the binary decision variables so, variable subscripts are used as the binary variables. In our model, two variable subscripts are used; s_w and k_{ε} .
- During implementing the CP code by using OPL software, some commands can be used as well by the following form under constraint programming fundamentals.
- $g_{k_{\varepsilon} o_{w'} s_{w'}}$ variable will be defined as Duration $[k, w, o, s]$.
- The variable $t_{k_{\varepsilon} o_{w'} s_{w'}}$ can be formulated by constraint programming as task $[k, \varepsilon, w, o, s].start$

train k finishes it, or trains k' and k are processed on section s on segment w and train k' precedes train k where train k can't use this segment before train k' finishes it.

$$t_{k_\varepsilon d_{s_w}} \geq t_{k_\varepsilon o_{s_w}} + \sum_{o=1}^{Q_w} \sum_{s=1}^{S_w} g_{k_\varepsilon o_{s_w}} \vee t_{k_\varepsilon d_{s_w}} \geq t_{k_\varepsilon o_{s_w}} + \sum_{o=1}^{Q_w} \sum_{s=1}^{S_w} g_{k_\varepsilon o_{s_w}} \quad (6)$$

$\forall k, k' \in K, k \neq k', s \in S_w, w \in E, \varepsilon, \varepsilon' \in R, o, o' \in Q$

Makespan Constraint

The following equation ensures each finish each operation for all trains during all runs has to be less than Makespan: see equation (7)

$$\max(o \text{ in operations}) (t_{k_\varepsilon o_{s_w}} + g_{k_\varepsilon o_{s_w}}) = C_{\max} \quad (7)$$

$\forall k \in K, s \in S, w \in E, o \in O, \varepsilon \in R.$

Train Constraints

Equations (8) and (9) are train capacity constraints. In equation (8), the number of empty wagons delivered for all sidings during run ε has to be less than or equal to the capacity of train k .

$$\sum_{w=1}^k \sum_{o=1}^O \sum_{s=1}^{S_w} \alpha_{k_\varepsilon o_{s_w}} \leq C_{K_\varepsilon} \quad \forall k, \varepsilon. \quad (8)$$

In equation (9), the number of full wagons collected from all sidings during run ε has to be less than or equal to the capacity of train k .

$$\sum_{w=1}^k \sum_{o=1}^O \sum_{s=1}^{S_w} B_{k_\varepsilon o_{s_w}} \leq C_{K_\varepsilon} \quad \forall k, \varepsilon. \quad (9)$$

Equation (10 - 11) ensures the delivering has to be in the outbound direction where o and o' are two operations at each siding in the two directions and o' precedes o , so:

If $o' \leq o$ then

$$\alpha_{k_\varepsilon o'_{s_w}} \leq C_{K_\varepsilon} \quad \forall k, \varepsilon, o', w, s. \quad (10)$$

$$B_{k_\varepsilon o'_{s_w}} = 0 \quad \forall k, \varepsilon, o', s, w. \quad (11)$$

Else

If $o' > o$ then

$$B_{k_\varepsilon o'_{s_w}} \leq C_{K_\varepsilon} \quad \forall k, \varepsilon, o', s, w. \quad (12)$$

- C_k capacity of train k of full wagons.
- $\varepsilon, \varepsilon'$ index of runs of each train; $\varepsilon = 1, 2, 3, \dots, R$, $\varepsilon' = 1, 2, 3, \dots, R$.
- $t_{k,o,w,s}$ start time of train k in run ε on section s on segment w .
- $g_{k,o,w,s}$ processing time of operation o of train k on section s on segment w .
- C_{max} makespan

$$\text{Minimise } C_{max} \quad (1)$$

Equation (2) ensures that operation $(o+1)$ of train k cannot be processed before finishing operation o for train k on section s of segment w .

$$t_{k,o,w,s} + g_{k,o,w,s} \leq t_{k,(o+1),w,s} \quad \forall k, \varepsilon, o, s, w. \quad (2)$$

Segments Order

Equation (3) ensures segment w is processed before segment w' or, segment w' before segment w

$$(t_{k,o,w,s} + g_{k,o,w,s}) \leq t_{k,o,w',s} \vee (t_{k,o,w',s} + g_{k,o,w',s}) \leq t_{k,o,w,s} \quad (3)$$

$\forall k, w, w'; w \neq w', \varepsilon, o, s.$

Selecting Run

Equation 4 ensures that train k selected run ε (unary resource) to be processed.

$$\text{ActivityHasSelect Resource } (k, \text{Trip}, \varepsilon) \Leftrightarrow k_{\varepsilon} \quad (4)$$

$\forall k \in K, \varepsilon \in R.$

Runs Order

Equation (5) ensures that run ε is assigned for train k before run ε' , or run ε' assigned for train k before run ε .

$$t_{k,o,s,\varepsilon} + g_{k,o,s,\varepsilon} \leq t_{k,o,s,\varepsilon'} \vee t_{k,o,s,\varepsilon'} + g_{k,o,s,\varepsilon'} \leq t_{k,o,s,\varepsilon} \quad (5)$$

$\forall k \in K, s, s' \in S, o \in O, w \in E, \varepsilon, \varepsilon' \in R; \varepsilon \neq \varepsilon'.$

Passing Priorities Constrains

Equation (6) ensures trains k and k' are processed on section s of segment w . train k precedes train k' where train k' can't use this segment before

Modelling any problem using CP depends on the CP package used because of the differences in constructs available in various modelling languages. In this study, ILOG's OPL modelling language will be used. OPL has a set of constructs to scheduling problems to develop an effect CP model on the scheduling problems. The basic OPL modelling framework includes a set of commands that can model the elements of scheduling problems. The segments and sections are the resources, and they are modelled as *unary resources* in OPL. Each run is a unary resource because it is assigned for one train as well. The characteristics of the unary resources are that they can process only one operation at the same time. Each operation is associated with a start time and duration. In addition, *ActivityHasSelectResource* is used in OPL to decide which unary resource will be selected for processing operation o of train k and using this command to know which run r will be selected by train k to start first. Also, each operation *requires* a unary resource to be processed on it. In this section we will provide a summary for a sugarcane transport system model by CP in the OPL language environment.

The main model contains two main parts: the objective function and the constraints. The model was formulated as a scheduling model using a constraint programming approach. The main advantage of the scheduling model is reactivity, where the mine and port requirements for coal wagons are met throughout the modelling period. In addition, the accuracy of the results and train passing are considered in this model.

Notations

K	number of trains.
k, k'	index of trains; $k=1,2,\dots,K$, $k'=1,2,\dots,K$.
E	maximum number of segments.
w	index of the segments; $w=1,2,\dots,E$.
S_w	total number of sections at segment w .
s_w	number of sections of segment w ; $s_w = 1_w, 2_w, \dots, S_w$.
$B_{k,o,w}$	number of full wagons collected from siding s by train k during operation o and run ε on segment w .
$\alpha_{k,o,w}$	number of empty wagons delivered for siding s by train k during operation o and run ε on segment w .
A_s	total allotment of siding s per day.
C_k	capacity of train k of empty wagons.

problems. A constraint programming approach was proposed for solving railway scheduling problems and obtaining solutions for many cases. (Rodriguez ,2007& Guel et al. 2009& Su Yun et al. 2002).

Masoud et al. (2010) developed a new mathematical model for sugarcane rail transport systems. In the sugarcane rail transport system, trains supply empty wagons to harvesters in the field and collect the filled wagons and return them to the mill for processing. The model considered many issues like trains scheduling, satisfying the mill and the harvester's requirements for empty and full wagons, satisfying the safety conditions by applying section blocking constraints and taking into account the capacities of trains, mill, and sidings which are usually limited. This model is generic and flexible and can be applied with minor alterations to other applications such as coal mining rail transport systems. Many features of sugarcane rail transport systems and coal mining rail transport systems are similar.

In this paper, the constraint programming (CP) approach is used to obtain optimal solutions for research problems. CP has many advantages such as: CP has the ability to model many different types of combinatorial optimisation problems while mathematical programming techniques individually have restrictive modelling power. It can produce global optimal solutions. Many problems in real life need multiple solutions; CP techniques have the capabilities to generate many solutions. CP techniques have the ability to solve all the feasibility problems and can deal with conflicting objectives,(Janiak et al. 2013).

THE MODEL

In this model, the constraint programming approach is used to formulate the coal rail transport system as an adapted job shop scheduling problem. The model defines the track sections as machines and the train activities as jobs. Each train activity (job) contains a number of operations defined by the start time and the processing time in a particular track section (machine). Some sections contain several parallel track sections, including sidings. The train trip from the port to the sidings and back to the port is called a run; see figure1. Each train typically completes several runs per day. The total time for all operations during a run is equal to the run time. The total run time per day is the total completion time. The main objective of this model has been minimising the completion times of the last operation in the system for all trains or Makespan.

Sidings have a finite capacity where the empties and fulls are delivered and collected that cannot be exceeded. Each siding has a daily allotment of wagons that often exceeds the capacity of the siding where the total delivers, or total collections, have to equal the siding allotment. Each train can haul a limited number of empty wagons and full wagons; this number depends on the capacity of each train. For safety conditions, the train is not allowed to have a mix of empties and fulls. As a result, all empties must take place before fulls.

The transport sector has a big impact on the overall costs of a coal mining production system. The total cost of the coal mining transport operations is very high; most of the total cost of coal mining production in Australia is incurred in coal transport. Some of the potential effects of mining transport on the production system are delaying the arrival of coal to the port; delaying the arrival of empty coal wagons to mines causing the mines to wait for empty wagons and increasing their costs; and increasing costs through inefficiencies in the coal mining transport system itself through the need for higher levels of labour and larger numbers of trains and coal wagons.

The coal rail transport system is very complex. Producing efficient schedules for coal mining transport can reduce the cost and limit the negative effects that this system can have on the raw mining production system. A railway system is very complicated, especially where there is only a single track. There are many studies and researchers have discussed and solved many problems in this sector to improve the efficiency of the system's performance. Most of studies consider that single track rail is an NP-hard problem and so needs special techniques to find a solution, (Burdett and Kozan(2010)) and (Kozan and Liu(2012))

Burdett and Kozan (2009, 2014) developed a new constructive technique to solve train scheduling problems using a job shop approach. The solution was obtained from a disjunctive graph model and constructive solution methods. This technique achieved better results than meta-heuristics techniques for different criteria such as minimizing a makespan objective.

Liu and Kozan (2009) represented the train scheduling problem as a blocking parallel machine job shop scheduling problem. In this model, they solved blocking problems in single and multiple railway tracks. They extended the classical disjunctive techniques and proposed a new heuristic method named the feasibility satisfaction procedure to solve this problem. The new technique treated many complex real situations in train crossing

A Constraint Programming Approach to Optimize Coal Mining Rail Operations

Mahmoud Masoud*

ABSTRACT

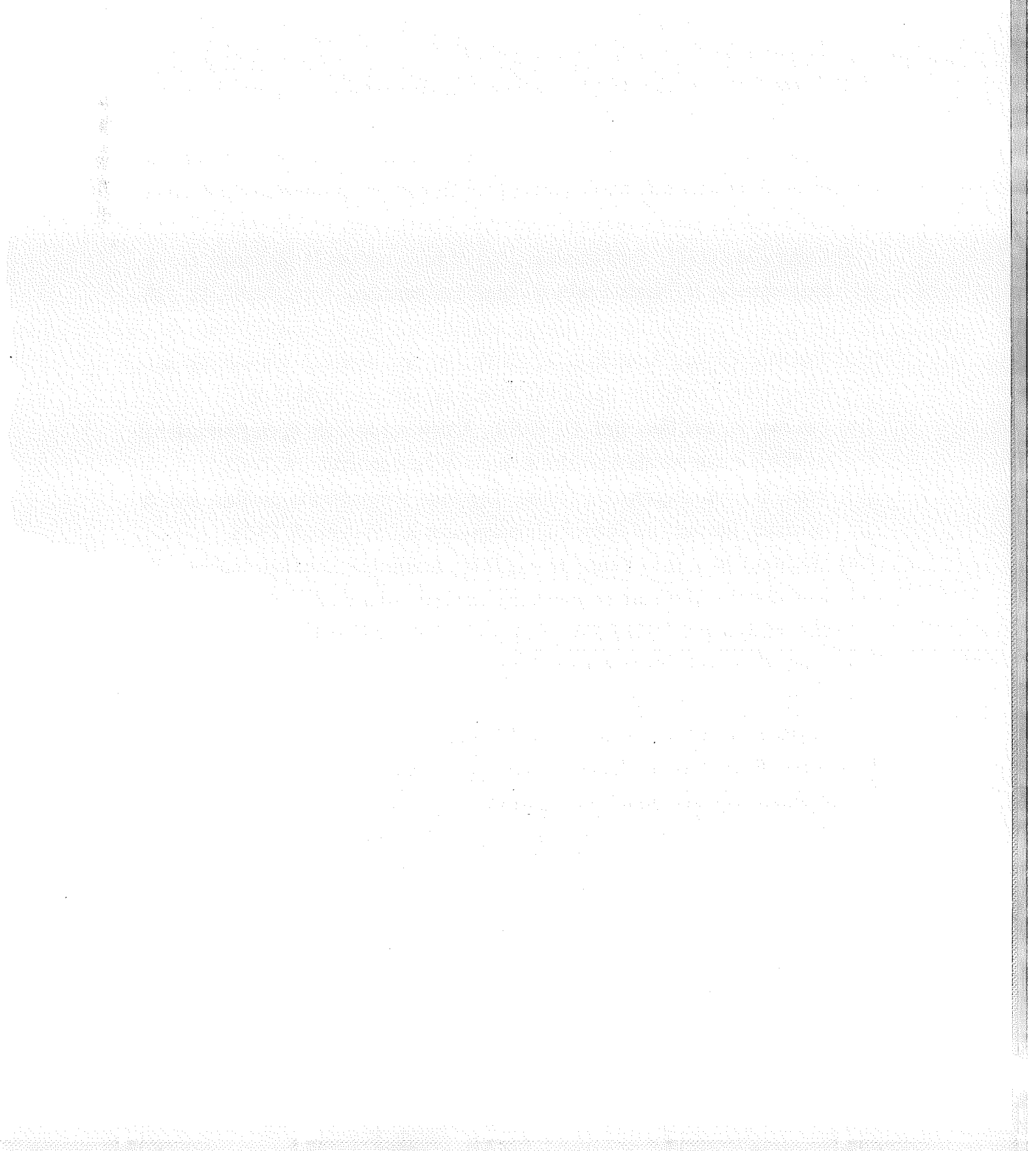
This paper presents a new model applied to produce efficient schedules by solving the coal transport scheduling problem. A coal rail system is very complicated, especially where it consists of a single track. Most studies consider single track rail to be an NP-hard problem, and so needs special techniques to find solutions. A job shop scheduling problem is adapted and formulated using a constraint programming approach to produce global optimal solutions. The new model includes the requirements of all major elements of the coal transport system to reduce the overall cost and optimise the performance of the system. Optimization Programming Language (OPL) was used to search for possible solutions to problems and for the best (optimal) solution based on specific criteria. The case study demonstrates that high quality solutions are obtainable for industry-scale applications.

Key Words: scheduling; rail transportation, mining.

INTRODUCTION

The coal mining transport system is an important element in the coal production system. Railways play a vital role for many countries such as Australia in transporting the coal from mines to ports, where the majority of the coal is transported by rail. A coal railway network uses a single track and includes many branch lines or segments and a lot of sidings. The coal mining railway system performs two main tasks: delivering empty wagons from the port to the mines at sidings; and collecting the full wagons of coal from mines and transporting them to the port.

*Assistant Prof. at Institute of Statistical Studies and Research – Cairo University
EMAIL: mailmasod@yahoo.com



- [9] Jimenez, F., and Verdegay, J. L., (1996), Interval multiobjective solid transportation problem via genetic algorithms, *Management of Uncertainty in Knowledge Based Systems, II*, 787-792.
- [10] Kassem, M. A., (1998), Stability of possibilistic multiobjective nonlinear programming problems without differentiability, *Fuzzy Sets and Systems*, 94, 239-246.
- [11] Li, Y., Ida, K., Gen, M., and Kobuchi, R., (1997), Neural network approach for multicriteria solid transportation problem, *Computer and Industrial Engineering*, 22, 465-468.
- [12] Luhandjula, M. K., (1987), Multiple objective programming problems with possibilistic coefficients, *Fuzzy Sets and Systems*, 21, 135-145.
- [13] Ojha, A., Fas, B., Mondal, S., and Maiti, M., (2010), A solid transportation problem for item with fixed charge, Vehicle cost and price discounted varying charge using genetic algorithm, *Applied Soft Computing*, 10, 100-110.
- [14] Osman, M., (1977), Qualitative analysis of basic notions in parametric convex programming, II (Parameters in the objective function). *Appl., Math.*, 22, 333-348.
- [15] Pandian, P., and Anuradha, D., (2010), A new approach for solving solid transportation problems, *Applied Mathematical Sciences*, 4, 3603-3610.
- [16] Pandian, P., and Natarajan, G., (2010), A new method for finding an optimal solution for transportation problems, *Engineering Application*, 4, 59-65.
- [17] Patel, G., and Tripathy, J., (1989), The solid transportation problem and its variations, *International Journal of management and Systems*, 5, 17-36.
- [18] Sakawa, M., and Yano, H., (1989), Interactive decision making for multiobjective nonlinear programming problems with fuzzy parameters, *Fuzzy Sets and Systems*, 29, 315-326.
- [19] Shell, E., (1955), Distribution of a product by several properties, Directorate management analysis, *Proceedings of the second symposium in linear programming*, DCS/Comptroller H. Q. U. SA. F., Washington, S. C., 2, 615-642.
- [20] Steuer, E. R., (1986). *Multiple criteria Optimization; Theory, Computation and Application* (Wiley, New York) 120-132.
- [21] Vajda, S., (1988), *Readings in linear programming*, Pitmen, London.
- [22] Zadeh, L. A., (1978), Fuzzy sets as a basis for a theory of possibility, *FSS* 1, 3-28.

$$z^{\min} = 237,$$

and

$$S(x^*) = \{w \in R^2 : 2w_1 + w_2 \geq 0, w_1 + 5w_2 \geq 0, w_1, w_2 \geq 0, w_1 + w_2 = 1\}.$$

9. Conclusion

The main objective of this paper is to present a solution procedure for multiobjective solid transportation problem with possibilistic variables (Poss MOSTP). A multi-objective solid transportation problem with possibilistic objective functions coefficients, possibilistic supplies, possibilistic demands and possibilistic conveyances has been introduced. The relation between the α -possibly for α -PMOSTP problem has been given. A parametric analysis to characterize the set of all α -parametric efficient solution for α -PMOSTP problem has been given. An algorithm to determine the stability set of the first kind corresponding to one of the parametric efficient solution of α -PMOSTP problem has been presented. A numerical example has been included in the sake of the paper for illustration. However, WINQSB computer package has been used to obtain the results.

References

- [1] Ammar, E. E., and Youness, E. A., (2005), Study on multiobjective transportation problem with fuzzy numbers, Applied Math. And Comput., 166, 241-253.
- [2] Basu, M., Pal, B. B., and Kundu A., (1994), An algorithm for finding the optimum solution of solid fixed charge transportation problem, 31, 283-291.
- [3] Bazaraa, S. M., Jarvis, J. J., and Sherall, D. H., (1990), Linear Programming and Netwrk Flows (Wiley, New York) 294-299.
- [4] Bit, A. K., Biswal, M. P., and Alam, S. S., (1993), Fuzzy programming problem, Fuzzy Sets and Systems, 57, 183-194.
- [5] Chanas, s., Kolodziejck, W., and Machaj, A., (1994), A fuzzy approach to the transportation problem, Fuzzy Sets and Systems, 13, 211-221.
- [6] Haley, K. B., (1962), The solid transportation problem, Oper. Res., 10, 448-468.
- [7] Hussein, M. L., (1998), Complete solutions of multiple objective transportation problems with possibilistic coefficients, Fuzzy Sets and Systems, 93, 293-299.
- [8] Ida, K., Gen, M., and Li, Y., (1995), Solving multicriteria solid transportation problem with fuzzy numbers by genetic algorithm, European Congress on Intelligent Techniques and Soft Computing (EUFIT'gs), Aachen, Germany, 434-441.

$$x \in G = \{x \in R^{12} : x_{111} + x_{121} + x_{131} + x_{112} + x_{122} + x_{132} = 20 - 4\theta,$$

$$x_{211} + x_{221} + x_{231} + x_{212} + x_{222} + x_{232} = 16 + 4\theta,$$

$$x_{111} + x_{211} + x_{112} + x_{212} = 16 - 4\theta,$$

$$x_{121} + x_{221} + x_{122} + x_{222} = 7 + 4\theta,$$

$$x_{131} + x_{231} + x_{132} + x_{232} = 11 + 4\theta,$$

$$x_{111} + x_{211} + x_{121} + x_{221} + x_{131} + x_{231} = 22 + 4\theta,$$

$$x_{112} + x_{212} + x_{122} + x_{222} + x_{132} + x_{232} = 14 - 4\theta,$$

$$x_{ijk} \geq 0, i = 1, 2; j = 1, 2, 3; k = 1, 2; 0 \leq \theta \leq 1\}$$

$$\begin{aligned} \min z_2(\theta) = & (14 - 4\theta)x_{111} + (5 + 4\theta)x_{211} + (7 - 4\theta)x_{121} + (7 + 4\theta)x_{221} \\ & + (2 + 4\theta)x_{131} + (6 + 4\theta)x_{231} + (6 - 4\theta)x_{112} + (14 + 4\theta)x_{212} \\ & + (10 - 4\theta)x_{122} + (5 + 4\theta)x_{222} + (7 - 4\theta)x_{132} + (7 + 4\theta)x_{232} \end{aligned}$$

subject to

$$x \in G.$$

For $w^* = (w_1^*, w_2^*) = (0.4, 0.6)$ and $\theta = 0.5$.

$$\begin{aligned} (\text{STP}(w))' \quad \min & (11.6x_{111} + 9x_{211} + 14x_{121} + 8.2x_{221} + 11.6x_{131} + 5.6x_{112} \\ & + 11.6x_{212} + 10x_{122} + 10.6x_{222} + 5.8x_{132} + 9.8x_{232}) \end{aligned}$$

subject to

$$x_{111} + x_{121} + x_{131} + x_{112} + x_{122} + x_{132} = 18,$$

$$x_{211} + x_{221} + x_{231} + x_{212} + x_{222} + x_{232} = 18,$$

$$x_{111} + x_{211} + x_{112} + x_{212} = 14,$$

$$x_{121} + x_{221} + x_{122} + x_{222} = 9,$$

$$x_{131} + x_{231} + x_{132} + x_{232} = 13,$$

$$x_{111} + x_{211} + x_{121} + x_{221} + x_{131} + x_{231} = 24,$$

$$x_{112} + x_{212} + x_{122} + x_{222} + x_{132} + x_{232} = 12,$$

$$x_{ijk} \geq 0, i = 1, 2; j = 1, 2, 3; k = 1, 2.$$

The solution is

$$\begin{aligned} (x_{111}, x_{211}, x_{121}, x_{221}, x_{131}, x_{231}, x_{112}, x_{212}, x_{122}, x_{222}, x_{132}, x_{232}) = \\ (0, 2, 0, 9, 6, 7, 12, 0, 0, 0, 0, 0), \end{aligned}$$

Table 2. Continue.

$\sup(\tilde{c}_{112}^2) = 6 - 4\theta$	$\pi_{\tilde{c}_{112}^2}(6) = \pi_{\tilde{c}_{112}^2}(2) = 0$
$\sup(\tilde{c}_{212}^2) = 14 + 4\theta$	$\pi_{\tilde{c}_{212}^2}(14) = \pi_{\tilde{c}_{212}^2}(18) = 0$
$\sup(\tilde{c}_{122}^2) = 10 - 4\theta$	$\pi_{\tilde{c}_{122}^2}(10) = \pi_{\tilde{c}_{122}^2}(6) = 0$
$\sup(\tilde{c}_{222}^2) = 5 + 4\theta$	$\pi_{\tilde{c}_{222}^2}(5) = \pi_{\tilde{c}_{222}^2}(9) = 0$
$\sup(\tilde{c}_{132}^2) = 7 - 4\theta$	$\pi_{\tilde{c}_{132}^2}(7) = \pi_{\tilde{c}_{132}^2}(3) = 0$
$\sup(\tilde{c}_{232}^2) = 7 + 4\theta$	$\pi_{\tilde{c}_{232}^2}(7) = \pi_{\tilde{c}_{232}^2}(11) = 0$

Table (3). The supports of $\tilde{a}_i$, $\tilde{b}_j$ and $\tilde{e}_k$ and the corresponding possibly distributions.

Support ($\tilde{a}_i$)	$\pi_{\tilde{a}_i}(\cdot)$
$\sup(\tilde{a}_1) = 20 - 4\theta$	$\pi_{\tilde{a}_1}(20) = \pi_{\tilde{a}_1}(16) = 0$
$\sup(\tilde{a}_2) = 16 + 4\theta$	$\pi_{\tilde{a}_2}(16) = \pi_{\tilde{a}_2}(20) = 0$
Support ($\tilde{b}_j$)	$\pi_{\tilde{b}_j}(\cdot)$
$\sup(\tilde{b}_1) = 16 - 4\theta$	$\pi_{\tilde{b}_1}(16) = \pi_{\tilde{b}_1}(12) = 0$
$\sup(\tilde{b}_2) = 7 + 4\theta$	$\pi_{\tilde{b}_2}(7) = \pi_{\tilde{b}_2}(11) = 0$
$\sup(\tilde{b}_3) = 11 + 4\theta$	$\pi_{\tilde{b}_3}(11) = \pi_{\tilde{b}_3}(15) = 0$
Support ($\tilde{e}_k$)	$\pi_{\tilde{e}_k}(\cdot)$
$\sup(\tilde{e}_1) = 22 + 4\theta$	$\pi_{\tilde{e}_1}(22) = \pi_{\tilde{e}_1}(26) = 0$
$\sup(\tilde{e}_2) = 14 - 4\theta$	$\pi_{\tilde{e}_2}(14) = \pi_{\tilde{e}_2}(10) = 0$

At $\alpha = 0$, the corresponding 0-PMOSTP problem is

$$\begin{aligned} \min z_1(\theta) = & (9 + 4\theta)x_{111} + (14 - 4\theta)x_{211} + (18 + 4\theta)x_{121} + (10 - 4\theta)x_{112} \\ & + (7 - 4\theta)x_{212} + (11 + 4\theta)x_{122} + (18 - 4\theta)x_{222} + (5 + 4\theta)x_{212} \\ & + (11 + 4\theta)x_{232} \end{aligned}$$

subject to

Table 1. The supports of $\tilde{c}_{ijk}^1$ and the corresponding possibly distributions:

Support ($\tilde{c}_{ijk}^1$)	$\pi_{\tilde{c}_{ijk}^1}(\cdot)$
$\sup(\tilde{c}_{111}^1) = 9 + 4\theta$	$\pi_{\tilde{c}_{111}^1}(9) = \pi_{\tilde{c}_{111}^1}(13) = 0$
$\sup(\tilde{c}_{211}^1) = 14 - 4\theta$	$\pi_{\tilde{c}_{211}^1}(14) = \pi_{\tilde{c}_{211}^1}(10) = 0$
$\sup(\tilde{c}_{121}^1) = 18 + 4\theta$	$\pi_{\tilde{c}_{121}^1}(18) = \pi_{\tilde{c}_{121}^1}(22) = 0$
$\sup(\tilde{c}_{221}^1) = 9 - 4\theta$	$\pi_{\tilde{c}_{221}^1}(9) = \pi_{\tilde{c}_{221}^1}(5) = 0$
$\sup(\tilde{c}_{131}^1) = 13 + 4\theta$	$\pi_{\tilde{c}_{131}^1}(13) = \pi_{\tilde{c}_{131}^1}(17) = 0$
$\sup(\tilde{c}_{231}^1) = 7 + 4\theta$	$\pi_{\tilde{c}_{231}^1}(7) = \pi_{\tilde{c}_{231}^1}(11) = 0$
$\sup(\tilde{c}_{112}^1) = 10 - 4\theta$	$\pi_{\tilde{c}_{112}^1}(10) = \pi_{\tilde{c}_{112}^1}(6) = 0$
$\sup(\tilde{c}_{212}^1) = 7 - 4\theta$	$\pi_{\tilde{c}_{212}^1}(7) = \pi_{\tilde{c}_{212}^1}(3) = 0$
$\sup(\tilde{c}_{122}^1) = 11 + 4\theta$	$\pi_{\tilde{c}_{122}^1}(11) = \pi_{\tilde{c}_{122}^1}(15) = 0$
$\sup(\tilde{c}_{222}^1) = 18 - 4\theta$	$\pi_{\tilde{c}_{222}^1}(18) = \pi_{\tilde{c}_{222}^1}(14) = 0$
$\sup(\tilde{c}_{132}^1) = 5 + 4\theta$	$\pi_{\tilde{c}_{132}^1}(15) = \pi_{\tilde{c}_{132}^1}(9) = 0$
$\sup(\tilde{c}_{232}^1) = 2 + 4\theta$	$\pi_{\tilde{c}_{232}^1}(2) = \pi_{\tilde{c}_{232}^1}(6) = 0$

Table 2. The supports of $\tilde{c}_{ijk}^2$ and the corresponding possibly distributions:

Support ($\tilde{c}_{ijk}^2$)	$\pi_{\tilde{c}_{ijk}^2}(\cdot)$
$\sup(\tilde{c}_{111}^2) = 14 - 4\theta$	$\pi_{\tilde{c}_{111}^2}(14) = \pi_{\tilde{c}_{111}^2}(10) = 0$
$\sup(\tilde{c}_{211}^2) = 5 + 4\theta$	$\pi_{\tilde{c}_{211}^2}(5) = \pi_{\tilde{c}_{211}^2}(9) = 0$
$\sup(\tilde{c}_{121}^2) = 7 - 4\theta$	$\pi_{\tilde{c}_{121}^2}(7) = \pi_{\tilde{c}_{121}^2}(3) = 0$
$\sup(\tilde{c}_{221}^2) = 7 + 4\theta$	$\pi_{\tilde{c}_{221}^2}(7) = \pi_{\tilde{c}_{221}^2}(11) = 0$
$\sup(\tilde{c}_{131}^2) = 2 + 4\theta$	$\pi_{\tilde{c}_{131}^2}(2) = \pi_{\tilde{c}_{131}^2}(6) = 0$
$\sup(\tilde{c}_{231}^2) = 6 + 4\theta$	$\pi_{\tilde{c}_{231}^2}(6) = \pi_{\tilde{c}_{231}^2}(10) = 0$

8. Numerical example

Consider the following

$$\text{(Poss MOSTP)} \quad \min \tilde{z}_1 = \sum_{i=1}^2 \sum_{j=1}^3 \sum_{k=1}^2 \tilde{c}_{ijk}^1 x_{ijk},$$

$$\min \tilde{z}_2 = \sum_{i=1}^2 \sum_{j=1}^3 \sum_{k=1}^2 \tilde{c}_{ijk}^2 x_{ijk}$$

subject to

$$x \in G = \left\{ x, \in R^{12} : \sum_{j=1}^3 \sum_{k=1}^2 x_{1jk} = \tilde{a}_1, \sum_{j=1}^3 \sum_{k=1}^2 x_{2jk} = \tilde{a}_2, \right.$$

$$\sum_{i=1}^2 \sum_{k=1}^2 x_{i1k} = \tilde{b}_1, \sum_{i=1}^2 \sum_{k=1}^2 x_{i2k} = \tilde{b}_2, \sum_{i=1}^2 \sum_{k=1}^2 x_{i3k} = \tilde{b}_3,$$

$$\sum_{i=1}^2 \sum_{j=1}^3 x_{ij1} = \tilde{e}_1, \sum_{i=1}^2 \sum_{j=1}^3 x_{ij2} = \tilde{e}_2,$$

$$x_{ijk} \geq 0, i = 1, 2; j = 1, 2, 3; k = 1, 2 \left. \right\}$$

In this example, two objective are simultaneously satisfied: the first and second are the minimization of possibilistic transportation costs.

Also, the possibilistic variables $\tilde{c}_{ijk}^1$, $\tilde{c}_{ijk}^2$, $\tilde{a}_i$, $\tilde{b}_j$, and $\tilde{e}_k$ are characterized by possibility distributions $\pi_{\tilde{c}_{ijk}^1}(\cdot)$, $\pi_{\tilde{c}_{ijk}^2}(\cdot)$, $\pi_{\tilde{a}_i}(\cdot)$, $\pi_{\tilde{b}_j}(\cdot)$ and $\pi_{\tilde{e}_k}(\cdot)$, respectively.

The supports of the possibilistic variables $\tilde{c}_{ijk}^1$, $\tilde{c}_{ijk}^2$, $\tilde{a}_i$, $\tilde{b}_j$, and $\tilde{e}_k$ are [9, 13], [10, 14], [20, 24], [15, 19] and [22, 29], respectively. it is appropriate to characterize these supports by parametric functions beginning by the points of maximum possibility of $\tilde{c}_{ijk}^1$ and $\tilde{c}_{ijk}^2$. Hence, the parametric functions of θ to the supports for $0 \leq \theta \leq 1$ are:

Kuhn-Tucker necessary optimality conditions (7)–(16), and solve the resulted system.

Step 6: Determine the stability set of the first kind $S(x^*)$ according to the values of the Lagrange multipliers δ_i , γ_i and ξ_k .

Step 7: Set $\theta = (\theta^* + \varepsilon) \in [0, 1]$ and go to step 2.

Repeat the above procedure until the interval $[0, 1]$ is fully exhausted. Then stop.

The flow chart of the proposed algorithm can be characterized as in the following figure:

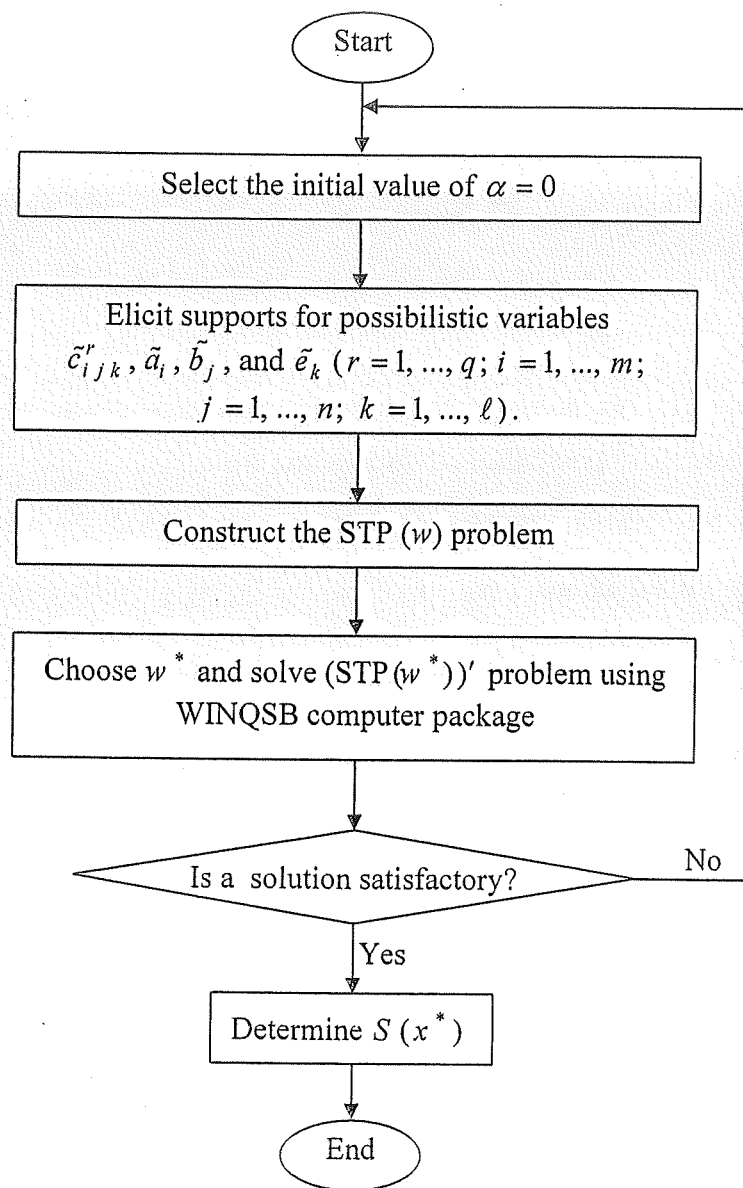


Fig. 1. Flowchart for proposed algorithm.

$$\sum_{j=1}^n \sum_{k=1}^{\ell} x_{ijk} = a_i^L(0) + \theta a_i^U(0), \quad i = 1, \dots, m \quad (9)$$

$$\sum_{i=1}^m \sum_{k=1}^{\ell} x_{ijk} = b_j^L(0) + \theta b_j^U(0), \quad j = 1, \dots, n \quad (10)$$

$$\sum_{i=1}^m \sum_{j=1}^n x_{ijk} = e_k^L(0) + \theta e_k^U(0), \quad k = 1, \dots, \ell \quad (11)$$

$$\sum_{i=1}^m (a_i^L + \theta a_i^U)_{\alpha=0} = \sum_{j=1}^n (b_j^L + \theta b_j^U)_{\alpha=0} = \sum_{k=1}^{\ell} (e_k^L + \theta e_k^U)_{\alpha=0} \quad (12)$$

$$\delta_i g_i(x, a_i) = 0, \quad i = 1, \dots, m \quad (13)$$

$$\gamma_j h_j(x, b_j) = 0, \quad j = 1, \dots, n \quad (14)$$

$$\xi_k f_k(x, e_k) = 0, \quad k = 1, \dots, \ell \quad (15)$$

$$\delta_i, \gamma_j \text{ and } \xi_k \geq 0 \quad (16)$$

where δ_i ($i = 1, \dots, m$), γ_j ($j = 1, \dots, n$), and ξ_k ($k = 1, \dots, \ell$) are the Lagrange multipliers and the above expressions are evaluated at x_{ijk}^* , c_{ijk}^{r*} , a_i^* , b_j^* and e_k^* . According to whether any of the variables δ_i ($i = 1, \dots, m$), γ_j ($j = 1, \dots, n$), and ξ_k ($k = 1, \dots, \ell$) are zero or positive, the stability of the first kind $S(x^*)$ of $(\text{STP}(w))'$ problem can be determined.

6. Proposed algorithm

The steps of the proposed algorithm to determine the stability set of the first kind $S(x^*)$ can be summarized as in the following steps:

Step 1: Start with an initial degree of $\alpha = 0$.

Step 2: Constrict $(\text{STP}(w))'$ problem.

Step 3: Ask the DM to specify the initial value of θ ($0 \leq \theta \leq 1$).

Step 4: Choose a certain $w^* \in B$ and solve $(\text{STP}(w))'$ problem using any available computer package (say, WINQSB Package). Let x^* be 0- parametric optimal solution of $(\text{STP}(w))'$ problem with the corresponding parameters $(c^{rL*}, c^{ru*}, a^{L*}, a^{U*}, b^{L*}, b^{U*}, e^{L*}, e^{U*})$.

Step 5: Substituting with x^* , c^{rL*} , c^{ru*} , a^{L*} , a^{U*} , b^{L*} , b^{U*} , e^{L*} , e^{U*} into the

Definition 9. Suppose that $w^* \in B$ with a corresponding α -parametric optimal solution $x^* \in E(w^*)$, then the stability set of the first kind of α -PMOSTP problem corresponding to x^* , denoted by $S(x^*)$ and is defined by:

$$S(x^*) = \{w \in B : x^* \in E(w^*) \text{ is an } \alpha\text{-parametric optimal solution of STP}(w) \text{ problem}\}.$$

For the sake of parametric analysis and for $\alpha = 0$, STP(w) problem can be written as in the following form (Bazara et al., (1990) and Steuer (1986))

$$(\text{STP}(w))' \min$$

$$\sum_{r=1}^q w_r z_r(x, c^r) = \min \sum_{r=1}^q \sum_{i=1}^m \sum_{m=1}^n \sum_{k=1}^{\ell} w_r (c_{ijk}^{rL}(0) + \theta c_{ijk}^{ru}(0)) x_{ijk}$$

subject to

$$g_i(x, a_i) = \sum_{j=1}^n \sum_{k=1}^{\ell} x_{ijk} - (a_i^L(0) + \theta a_i^U(0)) = 0, \quad i = 1, \dots, m;$$

$$h_j(x, b_j) = \sum_{i=1}^m \sum_{k=1}^{\ell} x_{ijk} - (b_j^L(0) + \theta b_j^U(0)) = 0, \quad j = 1, \dots, n;$$

$$f_k(x, e_k) = \sum_{i=1}^m \sum_{j=1}^n x_{ijk} - (e_k^L(0) + \theta e_k^U(0)) = 0, \quad k = 1, \dots, \ell;$$

$$\sum_{i=1}^m (a_i^L + \theta a_i^U)_{\alpha=0} = \sum_{j=1}^n (b_j^L + \theta b_j^U)_{\alpha=0} = \sum_{k=1}^{\ell} (e_k^L + \theta e_k^U)_{\alpha=0}$$

$$x_{ijk} \geq 0, \quad i = 1, \dots, m; \quad j = 1, \dots, n; \quad k = 1, \dots, \ell;$$

$$w \in W \text{ and } \theta \in [0, 1].$$

Let $w^* \in B$ with x^* is an α -parametric efficient solution of α -PMOSTP problem, then the Kuhn-Rucker necessary optimality conditions corresponding to (STP w)' problem will take the form:

$$\sum_{r=1}^q w_r \frac{\partial z_r}{\partial x_{\eta}} - \left(\sum_{i=1}^m \delta_i \frac{\partial g_i}{\partial x_{\eta}} + \sum_{j=1}^n \gamma_j \frac{\partial h_j}{\partial x_{\eta}} + \sum_{k=1}^{\ell} \xi_k \frac{\partial f_k}{\partial x_{\eta}} \right) = 0, \quad (7)$$

$$\sum_{r=1}^q w_r \frac{\partial z_r}{\partial \theta} - \left(\sum_{i=1}^m \delta_i \frac{\partial g_i}{\partial \theta} + \sum_{j=1}^n \gamma_j \frac{\partial h_j}{\partial \theta} + \sum_{k=1}^{\ell} \xi_k \frac{\partial f_k}{\partial \theta} \right) = 0, \quad (8)$$

From (4) and (6) we arrive to the contradiction of the efficiency of $x^*(c)$ for α -PMOSTP at certain $\alpha \in [0, 1]$.

Problem (α -PMOSTP) will be treated using the weighting approach (see, Chanas et al., (1984), i.e., defining the following problem (STP (w))

$$\begin{aligned} \text{(STP}(w)) \quad & \min \sum_{r=1}^q w_r z_r(x, c^r) \\ \text{subject to} \quad & x(c) \in G(a, b, e), c_{ijk}^r \in (\tilde{c}_{ijk}^r)_\alpha, r = 1, \dots, q; i = 1, \dots, m; \\ & j = 1, \dots, n; k = 1, \dots, \ell; a_i \in (\tilde{a}_i)_\alpha, i = 1, \dots, m; b_j \in (\tilde{b}_j)_\alpha, \\ & j = 1, \dots, n; e_k \in (\tilde{e}_k)_\alpha, k = 1, \dots, \ell, \sum_{i=1}^m (a_i)_\alpha = \sum_{j=1}^n (b_j)_\alpha = \\ & \sum_{k=1}^{\ell} (e_k)_\alpha, w \in W = \left\{ w \in R : \sum_{r=1}^q w_r = 1, w_r \geq 0 \right\}, \\ & i = 1, \dots, m; j = 1, \dots, n; k = 1, \dots, \ell \end{aligned}$$

It is clear that problem STP(w) is a single objective solid transportation problem which is considered as a parametric nonlinear programming problem since the possibilistic parameters will be treated as decision variables.

Definition 7. The set of α -parametric optimal solutions of STP(w) is defined as:

$$\begin{aligned} E(w) = \{x^* \in R^{q(m \times n \times \ell)} : \sum_{r=1}^q w_r z_r(x^*, c^{*r}) \leq \sum_{r=1}^q w_r z_r(x, c^{*r}), \text{ for each} \\ x(c^*) \in G(a^*, b^*, e^*), c^* \in (\tilde{c})_\alpha, a^* \in (\tilde{a})_\alpha, b^* \in (\tilde{b})_\alpha, e^* \in (\tilde{e})_\alpha \\ \text{and } w_r > 0, r = 1, \dots, q, \sum_{r=1}^q w_r = 1\}. \end{aligned}$$

Remark 1. A point x^* is said to be a proper α -parametric efficient solution of α -PMOSTP problem if and only if there exists $w^* > 0$ such that $x^* \in E(w^*)$.

5. Parametric analysis.

Definition 8. The solvability set of α -PMOSTP problem is defined by:

$$B = \{w \in R^q : \text{there exists } \alpha\text{-parametric efficient solution } x^* \text{ of } \alpha\text{-PMOSTP problem, } x^* \in E(w)\}.$$

$$\begin{aligned} \text{Poss } (z_1(x^1, \tilde{c}^1) \leq z_1(x^*, \tilde{c}^1), \dots, z_{r-1}(x^1, \tilde{c}^{r-1}) \leq z_{r-1}(x^*, \tilde{c}^{r-1}), \\ z_r(x^1, \tilde{c}^r) < z_r(x^*, \tilde{c}^r), z_{r+1}(x^1, \tilde{c}^{r+1}) \leq z_{r+1}(x^*, \tilde{c}^{r+1}), \dots, \\ z_q(x^1, \tilde{c}^q) \leq z_q(x^*, \tilde{c}^q) \geq \alpha, \alpha \in [0, 1]. \end{aligned}$$

This contradicts the α -possibly efficient of $x^*(c) \in G(a, b, e)$ for Poss MOSTP and the necessity part is established.

Sufficiency: Let $x^*(c) \in G(a, b, e)$ be α -parametric efficient solution for α -PMOSTP and $x^*(c) \in G(a, b, e)$ be not an α -possibly efficient solution for Poss MOSTP. Then there are $x^2 \in G(a, b, e)$ and $r = 1, \dots, q$ such that

$$\begin{aligned} \text{Poss } (z_1(x^2, \tilde{c}^1) \leq z_1(x^*, \tilde{c}^1), \dots, z_{r-1}(x^2, \tilde{c}^{r-1}) \leq z_{r-1}(x^*, \tilde{c}^{r-1}), \\ z_r(x^2, \tilde{c}^r) < z_r(x^*, \tilde{c}^r), z_{r+1}(x^2, \tilde{c}^{r+1}) \leq z_{r+1}(x^*, \tilde{c}^{r+1}), \dots, \\ z_q(x^2, \tilde{c}^q) \leq z_q(x^*, \tilde{c}^q) \geq \alpha, \end{aligned}$$

i.e.,

$$\begin{aligned} \sup_{(c^1, \dots, c^q) \in \bar{C}} \min(\pi_{\tilde{c}^1}(c^1), \dots, \pi_{\tilde{c}^{r-1}}(c^{r-1}), \pi_{\tilde{c}^r}(c^r), \pi_{\tilde{c}^{r+1}}(c^{r+1}), \dots, \\ \pi_{\tilde{c}^q}(c^q)) \geq \alpha, \end{aligned} \quad (4)$$

where

$$\begin{aligned} \text{Poss } \bar{C} = \{(c^1, \dots, c^q) \in R^{q(m \times n \times \ell)} : z_1(x^2, c^1) \leq z_1(x^*, c^1), z_{r-1}(x^2, c^{r-1}) \leq, \\ z_{r-1}(x^*, c^{r-1}), z_r(x^2, c^r) < z_r(x^*, c^r), z_{r+1}(x^2, c^{r+1}) \leq, \\ z_{r+1}(x^*, c^{r+1}), \dots, z_q(x^2, c^q) \leq z_q(x^*, c^q)\}. \end{aligned}$$

For the sup remum to be existed, there is $(P^1, P^2, \dots, P^q) \in \bar{C}$ with $\min(\pi_{\tilde{c}^1}(P^1), \pi_{\tilde{c}^2}(P^2), \pi_{\tilde{c}^q}(P^q)) < \alpha$, then

$$\sup_{(P^1, P^2, \dots, P^q) \in \bar{C}} \min(\pi_{\tilde{c}^1}(P^1), \pi_{\tilde{c}^2}(P^2), \pi_{\tilde{c}^q}(P^q)) < \alpha.$$

This contradicts (8). Then there is $(P^1, P^2, \dots, P^q) \in \bar{C}$ satisfying

$$\min(\pi_{\tilde{c}^1}(P^1), \pi_{\tilde{c}^2}(P^2), \pi_{\tilde{c}^q}(P^q)) \geq \alpha \quad (5)$$

i.e.,

$$P^r \in (\tilde{c}_{ijk}^r)_\alpha, \quad r = 1, \dots, q; \quad i = 1, \dots, m; \quad j = 1, \dots, n; \quad k = 1, \dots, \ell \quad (6)$$

$$\begin{aligned}
 & r = 1, \dots, q; a_i \in (\tilde{a}_i)_\alpha, i = 1, \dots, m; b_j \in (\tilde{b}_j)_\alpha, j = 1, \dots, n; \\
 & e_k \in (\tilde{e}_k)_\alpha, k = 1, \dots, \ell, \sum_{i=1}^m (a_i)_\alpha = \sum_{j=1}^n (\tilde{b}_j)_\alpha = \sum_{k=1}^{\ell} (\tilde{b}_j)_\alpha \\
 & = \sum_{k=1}^{\ell} (\tilde{e}_k)_\alpha, \text{ and} \\
 & x_{ijk} \geq 0, i = 1, \dots, m; j = 1, \dots, n; k = 1, \dots, \ell,
 \end{aligned} \tag{6}$$

where $(\tilde{c}_{ijk}^r)_\alpha, (\tilde{a}_i)_\alpha, (\tilde{b}_j)_\alpha$ and $(\tilde{e}_k)_\alpha$ denote the α -cut of the possibilistic variables $\tilde{c}_{ijk}^r, \tilde{a}_i, \tilde{b}_j$ and $\tilde{e}_k$, respectively. By the convexity assumption, $\pi_{\tilde{c}_{ijk}^r}(c_{ijk}^r), r = 1, 2, \dots, q; i = 1, \dots, m; j = 1, \dots, n; k = 1, \dots, \ell; \pi_{\tilde{a}_i}(a_i), i = 1, \dots, m; \pi_{\tilde{b}_j}(b_j), j = 1, \dots, n; \pi_{\tilde{e}_k}(e_k), k = 1, \dots, \ell$ are real intervals that will be denoted as $[c_{ijk}^{r-}(\alpha), c_{ijk}^{r+}(\alpha)], [a_i^{r-}(\alpha), a_i^{r+}(\alpha)], [b_j^{r-}(\alpha), b_j^{r+}(\alpha)], [e_k^{r-}(\alpha), e_k^{r+}(\alpha)]$.

Definition 7. A point $x^*(c) \in G(a, b, e)$ is said to be α -parametric efficient solution for α -PMOSTP found only if there $x \in G(a, b, e)$ and $c_{ijk}^r \in (c_{ijk}^r)_\alpha$ such that $z_r(x, c^r) \leq z_r(x^*, c^r)$ for all $r = 1, \dots, q$ and strict inequality holds for at least one r .

Theorem 1. A point $x^*(c) \in G(a, b, e)$ is an α -possibly efficient solution for Poss MOSTP if and only if $x^*(c) \in G(a, b, e)$ is an α -parametric efficient solution for α -PMOSTP.

Proof: (Necessity). Let $x^*(c) \in G(a, b, e)$ be an α -possibly efficient solution for Poss MOSTP and $x^*(c) \in G(a, b, e)$ be not an α -parametric efficient solution for α -PMOSTP. Then there are $x^1(c) \in G(a, b, e), t^r \in (\tilde{c}_{ijk}^r)_\alpha, r = 1, \dots, q; i = 1, \dots, m; j = 1, \dots, n; k = 1, \dots, \ell$ such that

$$z_s(x^1, t^s) \leq z_s(x^*, t^s), \text{ for all } s \in \{1, \dots, q\} \text{ and } r \in \{1, \dots, q\},$$

such that

$$z_r(x^1, t^r) < z_r(x^*, t^r).$$

As $t^r \in [\tilde{c}_{ijk}^r]_\alpha, r = 1, \dots, q$, we have

$$j = 1, 2, \dots, n;$$

and

$$\text{Poss} \left(\sum_{i=1}^m \sum_{j=1}^n x_{ijk} = \tilde{e}_k \right) = \sup(\pi_{\tilde{e}_k}(e_k)) \geq \beta_{3k}, \sum_{i=1}^m \sum_{j=1}^n x_{ijk} = e_k,$$

$$k = 1, 2, \dots, \ell,$$

where Poss denotes possibility.

Definition 6. (α -possibly efficient): A point $x^*(\tilde{c}) \in G(\tilde{a}, \tilde{b}, \tilde{e})$ is said to be α -possibly efficient to Poss MOSTP if there is no $x(\tilde{c}) \in G(\tilde{a}, \tilde{b}, \tilde{e})$ such that:

$$\begin{aligned} &\text{Poss}(z_1(x, \tilde{c}^1) \leq z_1(x^*, \tilde{c}^1), \dots, z_{r-1}(x, \tilde{c}^{r-1}) \leq z_{r-1}(x^*, \tilde{c}^{r-1}), \\ &z_r(x, \tilde{c}^r) < z_r(x^*, \tilde{c}^r), z_{r+1}(x, \tilde{c}^{r+1}) \leq z_{r+1}(x^*, \tilde{c}^{r+1}), \dots, \\ &z_q(x, \tilde{c}^q) \leq z_q(x^*, \tilde{c}^q)) \geq \alpha, \end{aligned} \quad (1)$$

On account of the extension principle,

$$\begin{aligned} &\text{Poss}(z_1(x, \tilde{c}^1) \leq z_1(x^*, \tilde{c}^1), \dots, z_{r-1}(x, \tilde{c}^{r-1}) \leq z_{r-1}(x^*, \tilde{c}^{r-1}), \\ &z_r(x, \tilde{c}^r) < z_r(x^*, \tilde{c}^r), z_{r+1}(x, \tilde{c}^{r+1}) \leq z_{r+1}(x^*, \tilde{c}^{r+1}), \dots, z_q(x, \tilde{c}^q) \\ &\leq z_q(x^*, \tilde{c}^q)) = \sup_{(c^1, c^2, \dots, c^q) \in C} \min(\pi_{\tilde{c}^1}(c^1), \pi_{\tilde{c}^2}(c^2), \dots, \pi_{\tilde{c}^{r-1}}(c^{r-1}), \\ &\pi_{\tilde{c}^r}(c^r), \pi_{\tilde{c}^{r+1}}(c^{r+1}), \dots, \pi_{\tilde{c}^q}(c^q)), \end{aligned} \quad (2)$$

where,

$$\begin{aligned} C = \{ &(c^1, c^2, \dots, c^q) \in R^{q(m \times n \times \ell)} : z_1(x, c^1) \leq z_1(x^*, c^1), \dots, z_{r-1}(x, c^{r-1}) \\ &\leq z_{r-1}(x^*, c^{r-1}), z_r(x, c^r) < z_r(x^*, c^r), z_{r+1}(x, c^{r+1}) \leq z_{r+1}(x^*, c^{r+1}), \\ &\dots, z_q(x, c^q) \leq z_q(x^*, c^q) \} \end{aligned} \quad (3)$$

For characterizing the α -possibly efficient solution for Poss MOSTP, let us consider the following α -parametric multiobjective solid transportation problem (α -PMOSTP)

$$(\alpha\text{-PMOSTP}) \quad \min z_r(x, c^r) = \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^{\ell} c_{ijk}^r x_{ijk}, r = 1, \dots, q \quad (4)$$

subject to

$$x \in G(a, e), c_{ijk}^r \in (\tilde{c}_{ijk}^r)_{\alpha}, i = 1, \dots, m; j = 1, \dots, n; k = 1, \dots, \ell;$$

3. Problem formulation.

Consider the general possibilistic multiobjective solid transportation problem with possibilistic coefficients in the objective functions, possibilistic supplies, possibilistic demands, and possibilistic conveyances (Poss MOSTP)

$$(\text{Poss MOSTP}) \min Z_r(x, \tilde{c}^r) = \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^{\ell} \tilde{c}_{ijk}^r x_{ijk}; r=1, \dots, q \quad (1)$$

subject to

$$\begin{aligned} x \in G(\tilde{a}, \tilde{b}, \tilde{e}) = & \left\{ x \in R^{m \times n \times \ell} : \sum_{j=1}^n \sum_{k=1}^{\ell} x_{ijk} = \tilde{a}_i; i=1, \dots, m; \right. \\ & \sum_{i=1}^m \sum_{k=1}^{\ell} x_{ijk} = \tilde{b}_j, j=1, \dots, n; \sum_{i=1}^m \sum_{j=1}^n x_{ijk} = \tilde{e}_k, \\ & k=1, \dots, \ell; \sum_{i=1}^m \tilde{a}_i = \sum_{j=1}^n \tilde{b}_j = \sum_{k=1}^{\ell} \tilde{e}_k, \text{ and} \\ & x_{ijk} \geq 0, i=1, \dots, m; j=1, \dots, n; \\ & \left. k=1, \dots, \ell \right\}, \end{aligned} \quad (2)$$

where, $\tilde{c}_{ijk}^r$ ($r=1, \dots, q$), $\tilde{a}_i$, $i=1, \dots, m$; $\tilde{b}_j$, $j=1, \dots, n$; $\tilde{e}_k$, $k=1, \dots, \ell$ are possibilistic variables on R characterized by possibility distributions $\pi_{\tilde{c}_{ijk}^r}$, $\pi_{\tilde{a}_i}$, $\pi_{\tilde{b}_j}$ and $\pi_{\tilde{e}_k}$, respectively. For all outcomes it is assumed that all possibility distributions involved in Poss MOSTP are convex ones with compact (i.e., bounded and closed) supports and $y_0 = \sup(y)$ (see, Definitions 2-4).

4. Efficiency aspects.

Definition 5. (α -possibly feasible actions): Let $\beta_1 = (\beta_{11}, \beta_{12}, \dots, \beta_{1m})$, $\beta_{1i} \in [0, 1]$, $i=1, 2, \dots, m$; $\beta_2 = (\beta_{21}, \beta_{22}, \dots, \beta_{2n})$, $\beta_{2j} \in [0, 1]$, $j=1, 2, \dots, n$; and $\beta_3 = (\beta_{31}, \beta_{32}, \dots, \beta_{3\ell})$, $\beta_{3k} \in [0, 1]$, $k=1, 2, \dots, \ell$. Then $x \in G(\tilde{a}, \tilde{b}, \tilde{e}) = \{x \in R^{m \times n \times \ell} : x_{ijk} \geq 0, i=1, 2, \dots, m; j=1, 2, \dots, n; k=1, 2, \dots, \ell\}$ is said to be α -possibly feasible actions for problem (Poss MOSTP) if:

$$\text{Poss} \left(\sum_{i=1}^m \sum_{k=1}^{\ell} x_{ijk} = \tilde{b}_j \right) = \sup(\pi_{\tilde{b}_j}(b_j)) \geq \beta_{2j}, \sum_{i=1}^m \sum_{k=1}^{\ell} x_{ijk} = b_j,$$

Poss MOSTP is presented. A numerical example is given to illustrate and clarify the obtained results.

2. Preliminaries.

In this section, we recall some definitions and lemma needed through the paper (Zadeh (1970)).

Definition 1. A possibilistic variable y on $T \subseteq R^n$ is a variable characterised by a possibility distribution on $\pi_y(t)$.

This means that, if y is a variable taking values in T , then a possibility distribution π_y associated with y may be viewed as a fuzzy constraint on the values that may be assigned to y . such a distribution is characterized by a possibility distribution function $\pi_y : T \rightarrow [0, 1]$ which associated with each $t \in T$ the degree of compatibility of the variable y with the realization $t \in T$.

If T is a Cartesian product of $T_1, T_2, \dots, T_n$, then $\pi_y(t_1, t_2, \dots, t_n)$ is an n -ary possibility distribution, i.e., $\pi_y(t) = (\pi_{y_1}(t_1), \pi_{y_2}(t_2), \dots, \pi_{y_n}(t_n))$.

Definition 2. The α -cut of a possibilistic variable y is $y_\alpha = \{t \in T : \pi_y(t) \geq \alpha\}$.

Definition 3. A possibility distribution π_y on T is said to be convex if

$$\pi_y(\lambda t^1 + (1-\lambda)t^2) \geq \min(\pi_y(t^1), \pi_y(t^2)), \quad \forall t^1, t^2 \in T, \lambda \in [0, 1].$$

It is appropriate to define the support of a possibilistic variable y in the case of $T \subseteq R^n$.

Definition 4. The support of a possibilistic variable y is

$$\text{sup}(y) = \left\{ t \in T : \sup_{u \in N_\varepsilon(t)} (\pi_y(u)) > 0 \text{ for all } \varepsilon > 0 \right\},$$

where, $N_\varepsilon(t) = \{u \in T : \|u - t\| < \varepsilon\}$.

Lemma 1. $\text{Sup}(y)$ is a closed set in T .

Proof. See, Hussein (1998).

existing solution methodologies, recently it is receiving the attention and the interest of the researchers in this field (Ido et al., 1995; and Brt et al., 1993). Haley (1962) introduced the solution procedure of STP which is an extension of the modified distribution method. Patel and Tripathy (1989) developed a computationally superior method for a STP with mixed constraints. Vajda (1988) proposed an algorithm for a multi-index transportation problem which is an extension of the modified distribution method. Basu et al. (1994) provided an algorithm for finding the optimum solution of solid fixed charge linear transportation problem. Li et al., (1997) designed a neural network approach for multicriteria STP. Jimenez and Verdegay (1996) developed a parametric approach for solving fuzzy STPs by an evolutionary algorithm (EA). Pandian and Natarajan (2010) introduced the zeropoint method for finding an optimal solution to a classical TP. Pandian and Auradha (2010) proposed a new method using the principle of zero point method introduced by Pandian and Natarajan (2010) for finding an optimal solution of STP. Ojha et al., (2010) formulated a STP with discounted costs, fixed charges and vehicle costs as a linear programming problem. Qualitative analysis of some basic notions such as the set of feasible parameters, the solvability set and the stability set of the first kind and the stability set of the second kind are introduced by Osman (1977). Luhandjula (1987) deals with multi-objective programming problems with possibilistic coefficients. Hussein (1998) introduced the complete solutions of multi-objective transportation problems with possibilistic coefficients. Sakawa and Yano (1989) provided the concept of α -Pareto optimality of fuzzy parametric programs. Kassem (1998) introduced the multi-objective nonlinear programming problems with possibilistic variables and without differentiability for the considered objective functions. Ammar and Youness (2005) introduced the solution of multi-objective TP with fuzzy objectives, fuzzy sources, and fuzzy destinations.

In this paper, we deal with a multi-objective solid transportation problem (Poss MOSTP) with possibilistic coefficients, possibilistic supply values, possibilistic demand values, and possibilistic conveyances. The concept of α -possibly efficient and α -parametric efficient solution are introduced. The relation between the above two concept of solutions is established. A parametric analysis is introduced to characterize the set of all α -parametric efficient solutions. An algorithm to determine the stability set of the first kind corresponding to one parametric efficient solution of

Study on Possibilistic Multiobjective Solid Transportation Problems

E. E. Ammar¹, M. L. Hussein², and H. A. Khalifa³

Abstract.

The solid transportation problem (STP) arises when bounds are given on three item properties. Usually, these properties are supply, demand, and type of product or mode of transport (conveyance). In this paper, the efficient solutions and stability of multiobjective solid transportation problem (Poss MOSTP) with possibilistic coefficients $\tilde{c}_{ijk}$ and / or possibilistic supply quantities $\tilde{a}_i$ and / or possibilistic demand quantities $\tilde{b}_j$ and / or possibilistic conveyances $\tilde{e}_k$ are investigated. We consider the problem by incorporating possibilistic data into the objective functions coefficients, supplies, demands and conveyances. The concept of α -possibly efficient is specified in which the ordinary efficient solution is extended based on the α -cut of a possibilistic variables. A solution of the weighting problem of Poss MOSTP is deduced. A necessary and sufficient condition for such a solution is established. The basic notions like the solvability set and the stability set of the first kind are defined and characterized that's to characterize the parametric optimal solution for the auxiliary problem. An algorithm for the determination of the stability set is proposed. Finally, a numerical example is given to illustrate the aspects of the developed results.

Keywords: Multi-objective solid transportation problems; Possibilistic variables; α -possibly efficient solution; α -possibly optimal solution; Stability notions.

1. Introduction

The solid transportation problem (STP) is a generalization of the well-known transportation problem (TP) in which three item properties (supply, demand and conveyance) are taken into account in the constraint set instead of two (supply and demand), and its of great use in public distribution systems. The STP was first stated by Shell (1955). Although STP was forgotten for long time, because of the new

1- Department of Mathematics, Faculty of Science, Tanta University, Tanta, Egypt.

2- Department of Mathematics, Faculty of Science, Kafrelsheikh University, Kafrelsheikh, Egypt.

3- Department of Operations Research, Institute of Statistical Studies and Research, Cairo University, Cairo, Egypt.

...the ... of ...

...the ... of ...

...the ... of ...

...the ... of ...

...the ... of ...

...the ... of ...

Decision Making, Proceeding of the Tenth International Conference: Expand and Enrich the Domain of Thinking and Application, Springen-Verlage, Berlin.

- [7] Orlovski, S., (1984), Multi-objective programming problems with fuzzy parameters, Control Cybernet, 13 (3) 175-183.
- [8] Osman, M., (1977), Qualitative analysis of the basic notions in parametric programming, II (Parameters in the objective function 1, Apl. Mat. (22) 333-348.
- [9] Osman, M., and El-Banna, A., (1993), Stability of multi-objective nonlinear programming problems with fuzzy parameters, Math. Comput. Simulation, (35): 321-326.
- [10] Pandian .P,(2013) , Multi-objective linear programming Problems involving fuzzy parameters,International Journal of Soft Computing and Engineering (IJSCE),3(2):2231-2307.
- [11] Pandian . P ,(2013) , Multi- objective programming approach for fuzzy linear programming problems , A applied Mathematical Sciences ,37(7):1811-1817.
- [12] Rockefellar, R., (1967), Duality and stability in extremal involving convex functions, Pacific. J. Math., (21): 167-181.
- [13] Sakawa, M., and Yano, H., (1989), Interactive decision making for multi-objective nonlinear programming problems with fuzzy parameters, Fuzzy Sets and Systems, (29): 315-326.
- [14] Some'K , Ulungu.B, Mohamed.H.I,and Some' .B ,(2011) A new method for solving nonlinear multi-objective optimization problems , JP Journal of Mathematical Sciences , 2(1):1-18.
- [15] Tanaka, H., and Asia, K., (1984), Fuzzy linear programming problems with fuzzy numbers, Fuzzy Sets and System, 3 (1): 1-10.
- [16] Veeramani.V , Duraisamy.C and Nagoogani.A,(2011), Solving fuzzy multi-objective linear programming problems with membership functions , Australian Journal of Basics and Applied Sciences,5(8):1163-1171.
- [17] Wang, X. M., Qin, Z. L., and Hu, Y. D., (2001), An interactive algorithm for solving multicriteria decision making: the attainable reference point method, IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 3 (31): 194-198.

$$S_{J_2}(1.8401, 0.5, 3, 6) = \{d_2 \in \mathbb{R}^2 : d_{21} = 3, d_{22} \geq 6\}.$$

For $J_3 = \{1\}$, $\beta_1 = 0$, $\beta_2 > 0$, then

$$S_{J_3}(1.8401, 0.5, 3, 6) = \{d_2 \in \mathbb{R}^2 : d_{21} \geq 3, d_{22} = 6\}.$$

For $J_4 = \{1, 2\}$, $\beta_1 > 0$, $\beta_2 = 0$, then

$$S_{J_4}(1.8401, 0.5, 3, 6) = \{d_2 \in \mathbb{R}^2 : d_{21} = 3, d_{22} = 6\}.$$

Hence

$$S(1.8401, 0.5, 3, 6) = \bigcup_{q=1} S_{J_q}(1.8401, 0.5, 3, 6)$$

5. Concluding remarks

In this paper, a combined interactive approach introduced by Megahed et al., (2011) has used for solving MONLP problem with fuzzy parameters in the objective functions. The proposed approach was based on the ARP method introduced by Wang et al., (2001) and the RD method introduced by Narula et al., (1994) and adopted the Lexicographic weighed Techebycheff program. The stability set of the first kind corresponding to the α -pareto optimal solution has been determined. However WINQSB compute package have been used to obtain the results.

References

- [1] Cohon, J. L., Cuffent, J. R., and Reville, C. S., (1990), An interactive approach to identify the best compromise solution for two objective \s shortest path problems, Computers & Operations research, 17 (2): 187-198.
- [2] Duboi, D., and Prade, H., (1980), Fuzzy Sets and Systems: Theory and Application (Academic Press, New York).
- [3] Kassem, M. A., and Ammar, E. E. (1996), A parametric study of multiobjective nonlinear programming problems with fuzzy parameters in the objective function, Fuzzy Sets and Systems, (8): 187-196.
- [4] Kaufman A., and Gupta, M. M., (1991), Fuzzy Mathematical Models in Engineering and Management Science (Elsevier Science Publishers, Amsterdam).
- [5] Megahed, A. A., El-Banna, A., youness, A. E., and Gomaa, H. G., (2011) A combined interactive approach for solving E-convex multi-objective non linear programming problem, Applied Mathematics and Computation, 2 (7): 6777-6784.
- [6] Narula, S. C., Kirilov, L., and Vassilev, V., (1993), An Interactive Algorithm for Solving Multiple Objective Nonlinear Programming Problems "Multiple Criteria

$$(x_1 - 3)^2 + (x_2 - 2)^2 \leq 4,$$

$$-2x_1 + x_2 \leq 0,$$

$$a_1 \in [1.4, 4.6], a_2 \in [3.8, 9.6],$$

$$0 \leq \alpha_i \in \mathbb{R}.$$

Hence

$$\text{lex min } \{ \delta_i, |x_1 + a_1 - 5.0019| + |x_2 + a_2 - 8.2668| \}, i = 1, 2$$

subject to

$$|x_1 + a_1 - 5.0019| \leq \delta_1,$$

$$|x_2 + a_2 - 8.2668| \leq \delta_2,$$

$$(x_1 - 3)^2 + (x_2 - 2)^2 \leq 4,$$

$$-2x_1 + x_2 \leq 0,$$

$$a_1 \in [1.4, 4.6], a_2 \in [3.8, 9.6],$$

$$0 \leq \delta_i \in \mathbb{R}.$$

The obtained solution is $(x^h, a^h) = (1.8401, 0.5, 3, 6)$,

$$f(x^h, a^h) = (4.8401, 6.5)$$

The first iteration:

$$(x^h, a^h) = (1.8401, 0.5, 3, 6),$$

$$f^h = (4.8401, 6.5), \bar{f}^0 = (8, 10)^t \text{ and } f^\vee = (5, 8.2668).$$

Is the solution satisfactory Y/N? Yes. Stop with $f(x^h, a^h) = \bar{f}(x^h, a^h)$

and $(x^h, a^h) = (x^*, a^*)$ as the final solution.

In order to reduce the computational effort through the determination of $S(1.8401, 0.5, 3, 6)$, we conform the left hand side of the following two intervals;

$1.4 \leq a_1 \leq 4.6$ and $3.8 \leq a_2 \leq 9.6$. Hence $S(1.8401, 0.5, 3, 6)$ is determined by applying the following condition: $\beta_1(3 - d_{21}) = 0, \beta_2(6 - d_{22}) = 0; \beta_1, \beta_2 \geq 0$

we have $J_1 \subseteq \{1, 2\}$. For $J_1 = \emptyset; \beta_1, \beta_2 = 0$, then

$$S_{J_1}(1.8401, 0.5, 3, 6) = \{d_2 \in \mathbb{R}^2 : d_{21} \geq 3, d_{22} \geq 6\}.$$

For $J_2 = \{1\}, \beta_1 > 0, \beta_2 = 0$, then

$$\delta_1 \leq x_1 + a_1,$$

$$\delta_2 \leq x_2 + a_2,$$

$$(x_1 - 3)^2 + (x_2 - 2)^2 \leq 0,$$

$$-2x_1 + x_2 \leq 0,$$

$$a_1 \in [1.4, 4.6], a_2 \in [3.8, 9.6],$$

$$0 \leq \delta_i \in \mathbb{R}.$$

Denote the corresponding criterion vector by $f^\vee$, the solution become $(x^\vee, a^\vee) = (3.0019, 7.2668)$, $f^\vee = f(x^\vee, a^\vee) = (5.0019, 8.2668)$.

Step 3 : Ask DM to specify an initial α -attainable reference point $f(a^0, x^0)$ such that

$$\bar{f}_i(x^0, a_i^0) > f_i(x^\vee, a_i^\vee) \text{ for every } i = 1, 2;$$

$$5.0019 < f_1(x^0, a_1^0) \leq 8? 8,$$

$$8.2668 < f_2(x^0, a_2^0) \leq 10? 10.$$

$$\text{Then } \bar{f}(x^0, a^0) = (8, 10)^t.$$

Step 4 : Calculate the weighting vector from the following relation

$$\bar{\lambda}_i = \frac{1}{\bar{f}_i^0 - f_i^\vee}, \quad \text{for every } i = 1, 2.$$

Then

$$\bar{\lambda}_1 = \frac{1}{\bar{f}_1^0 - f_1^\vee} = \frac{1}{2.9981}, \quad \bar{\lambda}_2 = \frac{1}{\bar{f}_2^0 - f_2^\vee} = \frac{1}{1.7332}.$$

Let $J = \{1, 2, \dots, k\}$, $J = J^0$, set the iteration counter $h = 0$ and go to step 5.

Step 5: Solve the following lexicographic weighted Tchebycheff programming problem with the calculating weights and $f^\vee$ as

$$\text{lex min } \{\delta_i, |f_1(x, a_1) - f_1^\vee| + |f_2(x, a_2) - f_2^\vee|\},$$

subject to

$$\bar{\lambda}_1 |f_1(x, a_1) - f_1^\vee| \leq \delta_1,$$

$$\bar{\lambda}_2 |f_2(x, a_2) - f_2^\vee| \leq \delta_2,$$

4. Numerical example

Consider the following problem.

$$\min ((x_1 + \tilde{a}_1), (x_2 + \tilde{a}_2))^t$$

subject to

$$(x_1 - 3)^2 + (x_2 - 2)^2 \leq 4, \quad -2x_1 + x_2 \leq 0, \quad x_1, x_2 \geq 0,$$

with membership functions

$$\mu_{\tilde{a}_i}(a_i) = \begin{cases} 0, & a_i \in (-\infty, a_i^1], \\ \frac{a_i - a_i^1}{a_i^2 - a_i^1}, & a_i \in [a_i^1, a_i^2], \\ 1, & a_i \in [a_i^2, a_i^3], \\ \frac{a_i^4 - a_i}{a_i^4 - a_i^3}, & a_i \in [a_i^3, a_i^4], \\ 0, & a_i \in [a_i^4, \infty). \end{cases}$$

$$\text{Let, } a_1^1 = 1, \quad a_1^2 = 2, \quad a_1^3 = 4, \quad a_1^4 = 5,$$

$$\text{and } a_2^1 = 3, \quad a_2^2 = 5, \quad a_2^3 = 9, \quad a_2^4 = 10$$

Step 1: Let the DM specify the initial value of $\alpha = 0.4$. Then from definition 3, we

$$\text{get } a_1 \in [(a_1^2 - a_1^1)\alpha + a_1^1, -(a_1^4 - a_1^3)\alpha + a_1^4]$$

$$\text{Hence, } a_1 \in [1.4, 4.6]. \quad \text{Similarly, } a_2 \in [(a_2^2 - a_2^1)\alpha + a_2^1, -(a_2^4 - a_2^3)\alpha + a_2^4]$$

Hence, $a_2 \in [3.8, 9.6]$. According to problem (3), we get

$$\min ((x_1 + a_1), (x_2, a_2))^T,$$

subject to

$$(x_1 - 3)^2 + (x_2 - 2)^2 \leq 4,$$

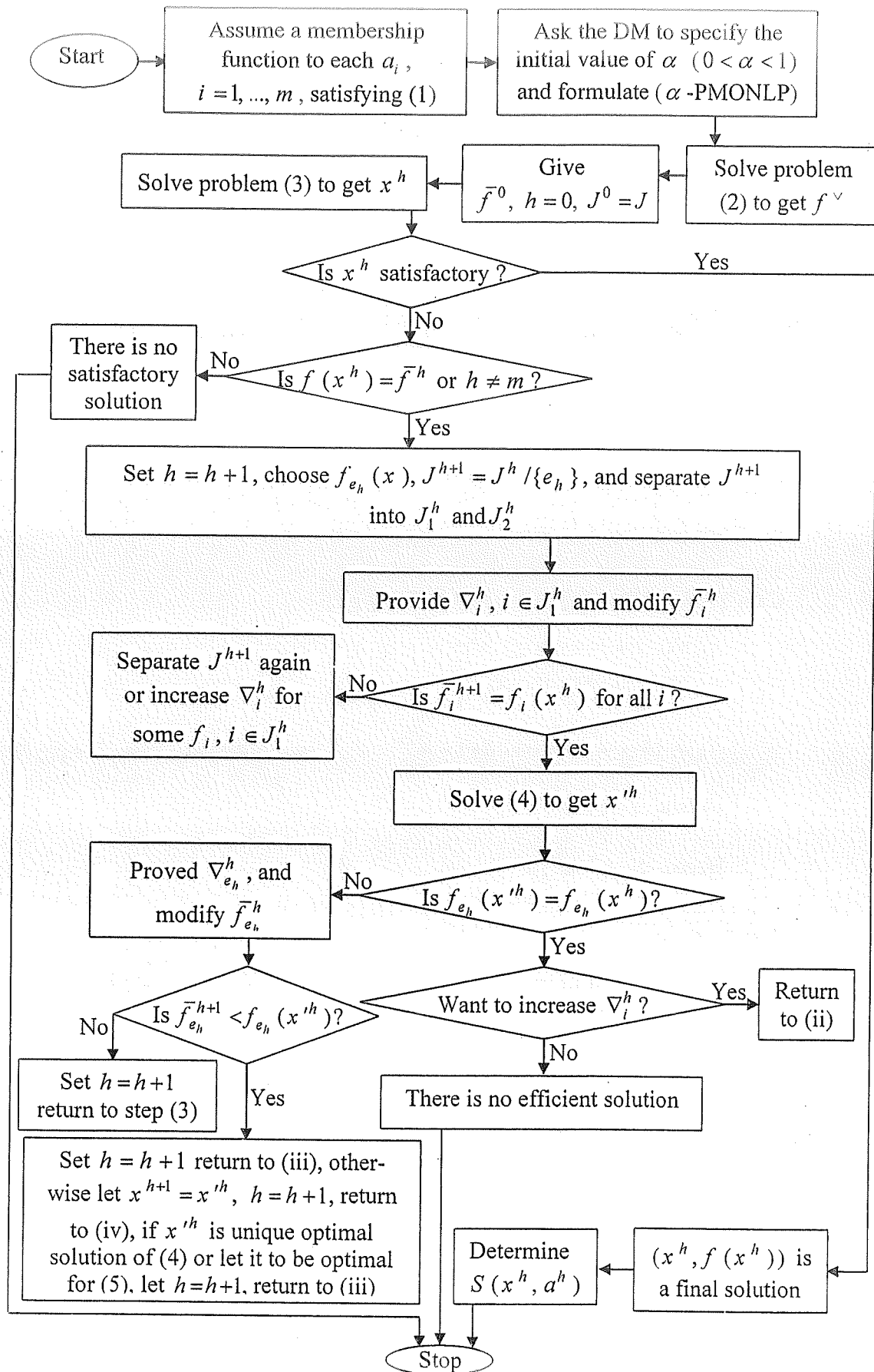
$$-2x_1 + x_2 \leq 0,$$

$$a_1 \in [1.4, 4.6], \quad a_2 \in [3.8, 9.6],$$

Step 2: Calculate starting point $(x^\vee, a^\vee)$ by solving the following problem

$$\max_{i=1,2} \delta_i$$

subject to



$$\min \delta_i, \quad i = 1, 2, \dots, m$$

subject to

$$\bar{\lambda}_i |f_i(x, a_i) - f_i^V| \leq \delta_i, \quad i = 1, \dots, m$$

$$, x \in M, a \in L_\alpha(\tilde{a}_i), 0 \leq \delta_i \in \square$$

let $h = h + 1$ and return to (iii). If $\bar{f}_{e_h}^{h+1} \geq f_{e_h}(x'^h, a_{e_h}'^h)$, let $h = h + 1$ and return to step (4).

Step 7: Determine $S(x^*, a^*)$ by applying the following conditions conditions:

$$\beta_i(a_i^* - d_{2i}) = 0, \quad i = 1, \dots, m,$$

$$\eta_i(d_{1i} - a_i^*) = 0, \quad i = 1, \dots, m,$$

$$\eta_i, \beta_i \geq 0, \quad i = 1, \dots, m.$$

Consider the following three cases:

Case 1: $\beta_i > 0, i \in I_1 \subset \{1, \dots, m\}, \beta_i = 0, i \notin I_1$.

$$\eta_i > 0, i \in I_2 \subset \{1, \dots, m\}, \eta_i = 0, i \notin I_2.$$

Let $N \subset \{1, \dots, m\}$ be the set of all proper subsets. Then

$$S_{I_1, I_2}(x^*, a^*) = \{(d_1, d_2) \in R^{2m} : d_{2i} = a_i^*, i \in I_1, d_{2i} > a_i^*, i \in I_1;$$

$$d_{1i} = a_i^*, i \in I_2, d_{1i} < a_i^*, i \notin I_2\};$$

$$\text{Hence, } S_1(x^h, a^h) = \bigcup_{I_1, I_2 \in N} S_{I_1, I_2}(x^h, a^h).$$

Case 2: $\beta_i, \eta_i = 0$, then

$$S_2(x^*, a^*) = \{(d_1, d_2) \in R^{2m} : d_{2i} > a_i^*, i = 1, \dots, m; d_{1i} \leq a_i^*, i = 1, \dots, m\}.$$

Case 3: $\beta_i, \eta_i > 0$, then

$$S_3(x^*, a^*) = \{(d_1, d_2) \in R^{2m} : d_{2i} = a_i^*, i = 1, \dots, m; d_{1i} = a_i^*, i = 1, \dots, m\},$$

Thus,

$$S(x^*, a^*) = \bigcup_{q=1}^3 S_q(x^*, a^*).$$

The following figure illustrates flowchart of the previous interactive approach.

(ii) For $i \in J_1^h$, DM provides $\nabla_i^h \in (0, \bar{f}_i^h - f_i(x^h, a_i^h)]$.

Let $\bar{f}_i^{h+1} = f_i(x^h, a_i^h) + \nabla_i^h$. For $i \in J_2^h$, let $\bar{f}_i^{h+1} = f_i(x^h, a_i^h)$ and for $i \in J^h / J^{h+1}$, let $\bar{f}_i^{h+1} = \bar{f}_i^h$.

(iii) In the case that $\bar{f}_i^{h+1} = f_i(x^h, a_i^h)$ for all $i \in J^h / \{e_h\}$, return to (i) to separate J^{h+1} a gain or return to (ii) to increase the amount to be relaxed for some f_i , $i \in J^h$ at $f(x^h, a^h)$ if DM wished to do so. Otherwise, stop and there is no satisfactory α -pareto optimal solution. In the case that $\bar{f}_i^{h+1} = f_i(x^h, a_i^h)$ for some $i \in J^h / \{e_h\}$, go to (iv).

(iv) Let $e = e_h$, $f_i' = \bar{f}_i^{h+1}$, $i = 1, \dots, m$, $i \neq e_h$, and solve the following auxiliary problem

$$\begin{aligned} & \min f_e(x, a_e) \\ & \text{subject to} \\ & f_i(x, a_i) \leq f_i', \quad i = 1, \dots, m; i \neq e, \\ & x \in M, a \in L_\alpha(\tilde{a}_i), \end{aligned} \tag{4}$$

to get a α -pareto optimal solution (x'^h, a'^h) . When

$f_{e_h}(x'^h, a_{e_h}'^h) = f_{e_h}(x^h, a_{e_h}^h)$ or $f_{e_h}(x'^h, a_{e_h}'^h)$ for f_{e_h} is not satisfactory to the DM, return to (ii) to increase the amount to be relaxed for some f_i ($i \in J_1^h$) at $f(x^h, a^h)$ if the DM wishes to do so, otherwise stop and there is no satisfactory α -pareto optimal solution. When $f_{e_h}(x'^h, a_{e_h}'^h) \neq f_{e_h}(x^h, a_{e_h}^h)$ and $f_{e_h}(x'^h, a_{e_h}'^h)$ for f_{e_h} is satisfactory to the DM, (he/she) provides ∇_e^h , the largest amount to be improved for f_{e_h} at $f(x^h, a^h)$, such that $\nabla_e^h \in (0, f_{e_h}(x^h, a_{e_h}^h) - f_{e_h}(x'^h, a_{e_h}'^h)]$.

Let $\bar{f}_{e_h}^{h+1} = f_{e_h}(x^h, a_{e_h}^h) - \nabla_e^h$.

(v) If $\bar{f}_{e_h}^{h+1} < f_{e_h}(x'^h, a_{e_h}'^h)$, let $h = h + 1$ and return to step (iii). Otherwise let $(x^{h+1}, a^{h+1}) = (x'^h, a'^h)$, let $h = h + 1$ and return to (iv) when (x'^h, a'^h) is a unique optimal solution of (4) or let (x'^h, a'^h) be an optimal solution of the following problem:

$$\begin{aligned} & \max \delta_i, \quad i = 1, \dots, m \\ & \text{subject to} \\ & f_i(x, a_i) \geq \delta_i, \quad i = 1, \dots, m; \\ & x \in M, a \in L_\alpha(\tilde{a}_i), i = 1, \dots, m; \delta_i \in \mathbb{R}. \end{aligned} \quad (2)$$

Step 3: DM provides an initial attainable reference point $\bar{f}_i^0$ such that $\bar{f}_i^0 > f_i^\vee$,
 $i = 1, \dots, m$. let $J = \{1, \dots, m\}$, $J^0 = J$, and the interaction counter $h = 0$.

Step 4: Calculate the weighting vector from the following relation:

$$\bar{\lambda} = \frac{1}{\bar{f}_i^h - f_i^\vee}, i = 1, \dots, m, \text{ and then Solve the following Lexicographic}$$

weighed Tchebycheff programming problem

$$\text{Let } \min \left(\delta_i, \sum_{i=1}^m \lambda_i |f_i(x, a_i) - f_i^\vee| \right)$$

subject to

$$\bar{\lambda}_i |f_i(x, a_i) - f_i^\vee| \leq \delta_i, i = 1, \dots, m \quad (3)$$

$$x \in M, a \in L_\alpha(\tilde{a}_i), i = 1, \dots, m,$$

to obtain an α -optimal solution (x^h, a^h) and the corresponding criterion vector $f(x^h, a^h)$.

Step 5: When $f(x^h, a^h)$ is satisfactory to the DM, let $(\bar{x}, \bar{a}) = (x^h, a^h)$ be the α -pareto optimal solution and go to step (7).

When $f(x^h, a^h)$ is not satisfactory and $f(x^h, a^h) = \bar{f}^h$ or $h = m$ there is no satisfactory α -pareto optimal solution of α -PMONLP problem. Otherwise go to step 6.

Step 6: Modify the reference point

- (i) The DM Chooses any e_h in J^h such that f_{e_h} is an unsatisfactory objective in $\{f_i : i \in J^h\}$ at $f(x^h, a^h)$. Let $J^{h+1} = J^h \setminus \{e_h\}$. Separate J^{h+1} into the following two parts: $J_1^h = \{i \in J^{h+1} : f_i(x^h, a^h) < \bar{f}_i^h\}$ and DM wishes to realize f_i at $f(x^h, a^h)$, and $J_2^h = J^{h+1} \setminus J_1^h$.

such that $f_i(x, a) \leq f_i(x^*, a^*)$, $i = 1, \dots, m$, with strict inequality holding for at least one i , where the corresponding values of the parameters $\tilde{a}$ are called α -level optimal parameters.

Definition 3. (Kaufman and Gupta (1991)). A trapezoidal fuzzy number (Tr.F.N.), $\tilde{A}$ can be represented completely by a quadruplet $\tilde{A} = (a_1, a_2, a_3, a_4)$.

We can characterize a Tr. F.N. by the interval of confidence at level α :

$A_\alpha = [(a_2 - a_1)\alpha + a_1, -(a_4 - a_3)\alpha + a_4]$; $\forall \alpha \in [0, 1]$. The membership function of a Tr. F.N. is characterized as

$$\mu_{\tilde{A}}(x) = \begin{cases} 0, & x < a_1, \\ \frac{x - a_1}{a_2 - a_1}, & a_1 \leq x \leq a_2, \\ 1, & a_2 \leq x \leq a_3, \\ \frac{a_4 - x}{a_4 - a_3}, & a_3 \leq x \leq a_4, \\ 0, & x \geq a_4, \end{cases}$$

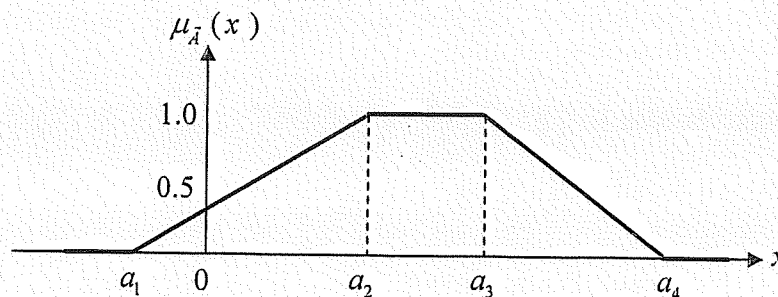


Fig. 1. A trapezoidal fuzzy number $\tilde{A} = (a_1, a_2, a_3, a_4)$.

3. A combined interactive approach

The steps of the approach to determine the α -pareto optimal solution (x^*, a_i^*) and their corresponding criterion vector $f_i(x^*, a_i^*)$, $i = 1, \dots, m$ can be summarized as in the following steps:

Step 1: Ask the decision maker (DM) to specify the initial value of α ($0 < \alpha < 1$) to formulate the (α -PMONLP) problem

Step 2: Find $f_i^\vee$ by solving the following problem

$$(FMONLP) \quad \min f_i(x, \tilde{a}_i)$$

subject to

$$x \in M = \{x \in R^n : g_j(x) \leq 0, j = 1, \dots, \ell\},$$

where $f : R^{n+m} \rightarrow R, i = 1, \dots, m; g_j(x) : R^n \rightarrow R, j = 1, \dots, \ell$ are convex differentiable functions and $\tilde{a}_i$ is a vector of fuzzy parameters. These fuzzy parameters are assumed to be characterized as the fuzzy numbers (Dubois and Prade (1980)). It is assumed that the feasible set of decision variables is nonempty compact (i.e., bounded and closed) and all the objectives are continuous.

It is appropriate to recall that the real fuzzy number $\tilde{a}$ from a convex continuous fuzzy subset of the real line whose membership function $\mu_{\tilde{a}}(a)$ is defined by:

- 1- A continuous mapping from R to the closed interval $[0, 1]$;
 - 2- $\mu_{\tilde{a}}(a) = 0$ for all $a \in (-\infty, a^1]$;
 - 3- strict increase on $[a^1, a^2]$;
 - 4- $\mu_{\tilde{a}}(a) = 1$ for all $a \in [a^2, a^3]$;
 - 5- strict decrease on $[a^3, a^4]$;
 - 6- $\mu_{\tilde{a}}(a) = 0$ for all $a \in [a^4, \infty]$.
- (1)

Definition 1. (Dubois and Prade (1980)). The α -level set of the fuzzy number $\tilde{a}$ is defined as the ordinary set $L_{\alpha}(\tilde{a})$ for which the degree of their membership functions exceeds the level α :

$$L_{\alpha}(\tilde{a}) = \{a : \mu_{\tilde{a}_i}(a_i) \geq \alpha, i = 1, \dots, m\}.$$

For a certain degree of α , the FMONLP problem can be written as in the following nonfuzzy (Rockafellar, (1967)).

$$(\alpha\text{-PMONLP}) \quad \min f_i(x, a_i)$$

subject to

$$x \in M, \quad a \in L_{\alpha}(\tilde{a})$$

Definition 2. $x^* \in M, a \in L_{\alpha}(\tilde{a})$ is said to be an α -pareto optimal solution to the problem α -PMONLP, if and only if there does not exist another $x \in M, a \in L_{\alpha}(\tilde{a})$

function exists, there is no a single optimal solution, but rather a set of non dominated or non inferior solutions from which a best compromise solution must be detected (Cohon et al., (1995)). In his earlier work, Osman (1977) introduced the notations of the solvability set, stability set of the first kind and stability set of the second kind, and analyzed these concepts for parametric convex nonlinear programming problems. Tanaka and Asai (1984) formulated multi-objective linear programming problems with fuzzy parameters. Orlovski (1984) formulated a general MONLP problem with fuzzy parameters. Sakawa and Yano (1989) introduced the concept of α -pareto optimality of MONLP problem with fuzzy parameters. Osman and El-banna (1993) introduced the stability of MONLP problems with fuzzy parameters. Kassem and Ammar (1996) introduced a parametric study of MONLP problems with fuzzy parameters in the objective functions.

Veeramani et al .,(2011), studied fuzzy multi-objective linear programming problem un which both technological coefficient and resources are fuzzy with linear membership function. Some' et al.,(2011) proposed a new method called multi-objective A linear Metaheuristec (MOMA) method for solving nonlinear multi-objective optimization problem. Pandian (2013) proposed a method to compute the solution for multi-objective fully fuzzy linear programming problem involving parameters represented by triangular fuzzy numbers. Pandian (2013) proposed a new method namely , level – sum method based on the multi-objective linear programming and the simplex method for computing an optimal fuzzy solution to a fuzzy linear programming problem in which fuzzy ranking functions are not used.

In this paper, a combined interactive approach introduced by Megahed et al., (2011) is used for solving MONLP problem with fuzzy parameters in the objective function. The proposed approach is based on the ARP method introduced by Wang et al., (2001) the RD method introduced by Narule et al., (1994) adopts the Lexicographic weighed Techebycheff program and the stability set of the first kind corresponding to the α -pareto optimal solution is determined.

2. Problem formulation

Consider the following multi-objective nonlinear programming problem (FMONLP) with fuzzy parameters in the objective functions :

An Interactive Approach for Solving Multi-objective Nonlinear Programming Problems with Fuzzy Parameters

E. E. Ammar¹, M. L. Hussein², and H. A. Khalifa³

Abstract

In general other is no single optimal solution in multi-objective problems, but rather a set of noninferior (pareto optimal) solutions from which the decision maker (DM) must select the most preferred or best compromise solution as the one to implement. In this paper, a combined interactive approach introduced by Megahed et al., (2011) is used for solving a multi-objective nonlinear programming problem (FMONLP) with fuzzy parameters in the objective functions. This approach is based on the attainable reference point (ARP) method introduced by Wang et al., (2001), the reference direction (RD) method introduced by Narula et al., (1994) and adopts the lexicographic weighted Techebyceff program. The fuzzy parameters are characterized by trapezoidal in the objective functions are characterized by trapezoidal fuzzy numbers. The concept of α -fuzzy efficient solution is specified in which the ordinary efficient solution is extended based on the α -level of fuzzy numbers. The stability set of the first kind without differentiability corresponding to α -efficient solution of α -PMONLP problem is determined. A numerical example is given in the sake of this paper to clarify the obtained results.

Keywords: Multi-objective nonlinear programming problems; Fuzzy parameters; Fuzzy numbers; α -pareto optimality; Attainable reference point method; Reference direction method; stability notions.

1. Introduction

Multi-objective analysis assumes that the objectives are generally in conflict. Therefore, unless a well defined utility function exists, there is no a defined utility

1- Department of Mathematics, Faculty of Science, Tanta University, Tanta, Egypt.

2- Department of Mathematics, Faculty of Science, Kafrelsheikh University, Kafrelsheikh, Egypt.

3- Department of Operations Research, Institute of Statistical Studies and Research, Cairo University, Cairo, Egypt.

2. The second part of the document focuses on the implementation of the proposed changes. It details the steps involved in the process, from the initial planning phase to the final execution. This section highlights the challenges faced during the implementation and the strategies used to overcome them. It also provides a timeline for the completion of the project, ensuring that all stakeholders are aware of the progress and can provide input as needed.

3. The third part of the document discusses the future of the organization. It outlines the long-term goals and the strategies to achieve them. This section also addresses the potential risks and the measures to mitigate them. It emphasizes the importance of continuous improvement and the need to stay abreast of the latest developments in the field. The document concludes with a statement of commitment to the organization's mission and vision, and a call to action for all stakeholders to work together to achieve the common goals.

4. The fourth part of the document provides a detailed analysis of the financial performance of the organization. It includes a breakdown of the revenue and expenses, and a comparison of the current performance with the previous year. This section also discusses the factors that have influenced the financial results, such as changes in market conditions and the effectiveness of the marketing strategy. It provides a forecast for the future financial performance, based on the current trends and the proposed changes. This section is intended to provide a clear picture of the financial health of the organization and to inform the decision-making process.

5. The fifth part of the document discusses the human resources of the organization. It outlines the current staffing levels and the skills of the employees. It also discusses the plans for recruitment and training, ensuring that the organization has the right people in the right positions. This section also addresses the issues of employee satisfaction and retention, and provides strategies to improve them. It emphasizes the importance of a strong and motivated workforce for the success of the organization.

6. The sixth part of the document discusses the legal and regulatory aspects of the organization's operations. It outlines the various laws and regulations that apply to the organization, and the measures taken to ensure compliance. This section also discusses the potential legal risks and the measures to mitigate them. It emphasizes the importance of a strong legal framework for the organization's operations and the need to stay up-to-date with the latest legal developments.

7. The seventh part of the document discusses the environmental and social aspects of the organization's operations. It outlines the organization's commitment to sustainability and the measures taken to reduce its carbon footprint. This section also discusses the organization's social responsibility initiatives, such as community development and employee welfare. It emphasizes the importance of a strong environmental and social performance for the long-term success of the organization.

8. The eighth part of the document provides a summary of the key findings and recommendations. It highlights the main points discussed in the document and provides a clear set of recommendations for the organization. This section is intended to provide a concise and easy-to-understand overview of the document's content and to guide the implementation of the proposed changes.

9. The ninth part of the document provides a list of references and a glossary of terms. The references include books, articles, and other sources used in the document. The glossary provides definitions for the key terms used in the document, ensuring that all readers have a clear understanding of the terminology. This section is intended to provide a comprehensive and easy-to-use reference for the document's content.

10. The tenth part of the document provides a list of appendices. These include detailed data tables, charts, and other supplementary information that supports the main text of the document. This section is intended to provide a comprehensive and easy-to-use reference for the document's content.

- [7] Osman M. S., Lashein E. F., Youness E.A. and Atteya T. E. M., Mathematical Programming in Rough Environment, Optimization: A Journal of Mathematical Programming and Operations Research, Vol 60. Issue 5, 603 – 611, 2011.
- [8] Osman M. S. A., Qualitative analysis of basic notions in parametric convex programming I (Parameters in the constraints), Applications of Mathematics, Vol. 22, No. 5, 318—332. 1977.
- [9] Osman. M. S. A., Qualitative analysis of basic notions in parametric convex programming. II (Parameters in the objective function), Applications of Mathematics, Vol. 22, No. 5. 333—348, 1977.
- [10] Pawlak Z., Rough sets, International Journal of Computer and Information Sciences (5). 341–356,1982.
- [11] Pawlak Z., Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Boston, 1991.
- [12] Pawlak Z., Rough Sets, Rough Relations and Rough Functions, Fundamental Inform. 27(2/3), 103-108, 1996.
- [13] Pawlak Z., Vagueness: A rough set view, Structures in Logic and Computer Science, Springer 106-117, 1997.
- [14] Pawlak Z., Rough set theory and its applications, Journal of telecommunications and information technology, 3/2002.
- [15] Tanino T., and Sawaragi Y., Duality theory in multiobjective programming, Journal of optimization theory and application. Vol 27, No 4, 1979.
- [16] Tao Z., and Xu. J., A class of rough multiple objective programming and its application to solid transportation problem, Information Sciences 188, 215–235, 2012.
- [17] Xu J., and Yao L., A class of multiobjective linear programming models with random rough coefficients, Mathematical and Computer Modelling 49, 189–206, 2009.
- [18] Yan Z., and Li S., Saddle point and generalized convex duality for multiobjective programming, Appl. & computing vol. 15, NO1-2, 227-235, 2004.
- [19] Youness E.A., Characterizing solutions of rough programming problems, European Journal of Operational Research, 168(3), 1019-1029, 2006.

Secondly: If $\ddot{D}_1 = \emptyset$, solve the following problem

$$\begin{aligned} & \text{Max} \sum_{i=1}^k w_i f_{*i}(\bar{x}) + \sum_{i=1}^k w_i (f_{*i}(x) - f_i^*(x)) + \sum_{r=1}^m \mu_r g_r(\bar{x}) \\ & \text{subject to} \sum_{i=1}^k w_i \nabla f_i(\bar{x}) + \sum_{i=1}^k w_i \nabla (f_{*i}(x) - f_i^*(x)) + \sum_{r=1}^m \mu_r \nabla g_r(\bar{x}) = 0 \\ & \mu_r \geq 0, r = 1, 2, \dots, m \end{aligned}$$

If the set $\ddot{D}_2$ is the optimal solution set and $\ddot{\theta}_*$ is the optimal value.

The set $\ddot{D} \cup \ddot{D}_2$ is a possibly optimal solution set.

4. Conclusion

In this paper we discuss the duality of the multiobjective rough convex programming problem when the multiobjective function is deterministic and roughness is in feasible region which does not discussed before. Also it presented the duality when roughness in multiobjective function and the feasible region is deterministic. The duality the multiobjective rough nonlinear convex programming problem is defined and its optimal solution sets are characterized. The duality study is important for these kinds of problems because it helps us for obtaining the surely optimal solution of the lower approximation function and the upper approximation function.

References

- [1] Abu-Donia H.M., Multi Knowledge based rough approximations and applications. Knowledge Based Systems 26, 20-29, 2012.
- [2] Baoding L., Theory and Practice of Uncertain Programming, 1st ed, Physica-Verlag, Heidelberg, 2002.
- [3] Bazaraa M. S., Sherali H., and Shetty C. M., Nonlinear Programming: Theory and Algorithms, 3rd ed, Wiley-Inter science, 2006.
- [4] Chankong V. and Hiams Y. Y., Multiobjective decision making: theory and methodology, New York: North Holland, 1983.
- [5] Feng, L., Li T., Ruan D., and Gou, .S., A vague-rough set approach for uncertain knowledge acquisition. Knowledge-Based Systems 24, 837-843, 2011.
- [6] LI Z. F and Wang S.Y., Lagrange multipliers and saddle point in multiobjective programming, Journal of optimization theory and application, Vol 83, No 1, 63-81, 1994.

Definition (The possibly Pareto optimal solution set):

A set $\ddot{C} \cup \ddot{C}_2$, $\ddot{C}_1 = \emptyset$, is called the possibly Pareto optimal solution set of the problem (7) which contains possibly Pareto optimal solutions.

Then the Pareto optimal set of the objective function $F(x)$ is between Pareto optimal set of the lower approximation functions vector $\ddot{F}_*$ and Pareto optimal set of the upper approximation functions vector $\ddot{F}^*$, $\ddot{F}_* \subseteq \min_{x \in X} F(x) \subseteq \ddot{F}^*$.

The dual problem of the second class of MRCPP is formulated as problem (5). So the optimal value of the dual problem of the second class is between

$$\ddot{\theta}_* \leq \max \ddot{\theta}(\mu) \leq \ddot{\theta}^*$$

where

$$\ddot{\theta}_* = \begin{cases} \max \sum_{i=1}^k w_i f_{*i}(\bar{x}) + \sum_{i=1}^k w_i (f_{*i}(x) - f_i^*(x)) + \sum_{r=1}^m \mu_r g_r(\bar{x}) \\ \text{subject to } \sum_{i=1}^k w_i \nabla f_i(\bar{x}) + \sum_{i=1}^k w_i \nabla (f_{*i}(x) - f_i^*(x)) + \sum_{r=1}^m \mu_r \nabla g_r(\bar{x}) = 0 \\ \mu_r \geq 0, r = 1, 2, \dots, m \end{cases}$$

$$\ddot{\theta}^* = \begin{cases} \max \sum_{i=1}^k w_i f_i^*(\bar{x}) + \sum_{i=1}^k w_i (f_{*i}(x) - f_i^*(x)) + \sum_{r=1}^m \mu_r g_r(\bar{x}) \\ \text{subject to } \sum_{i=1}^k w_i \nabla f_i(\bar{x}) + \sum_{i=1}^k w_i \nabla (f_{*i}(x) - f_i^*(x)) + \sum_{r=1}^m \mu_r \nabla g_r(\bar{x}) = 0 \\ \mu_r \geq 0, r = 1, 2, \dots, m \end{cases}$$

The dual problem can be solved in the following manner:

Firstly: Solve the following problem

$$\begin{aligned} & \max \sum_{i=1}^k w_i f_i^*(\bar{x}) + \sum_{i=1}^k w_i (f_{*i}(x) - f_i^*(x)) + \sum_{r=1}^m \mu_r g_r(\bar{x}) \\ & \text{subject to } \sum_{i=1}^k w_i \nabla f_i(\bar{x}) + \sum_{i=1}^k w_i \nabla (f_{*i}(x) - f_i^*(x)) + \sum_{r=1}^m \mu_r \nabla g_r(\bar{x}) = 0 \\ & \mu_r \geq 0, r = 1, 2, \dots, m \end{aligned}$$

If $\ddot{D}$ is the optimal solution set and $\ddot{\theta}^*$ is the optimal value.

Find the set $\ddot{D}_1 = \{\mu \in \ddot{D} | \ddot{\theta}_*(\mu) = \ddot{\theta}^*\}$ which is a surely optimal solution set if $\ddot{D}_1 \neq \emptyset$.

3.2 Dual Problem of Second Class of MRCPP:

The second class of MRCPP is formulated as follows:

$$\begin{aligned} & \min F(x) \\ & \text{subject to } x \in X \end{aligned} \quad (7)$$

where X is the feasible region such that $X = \{x \in R^n | g_i(x) \leq 0 \ i = 1, 2, \dots, m\}$, $F(x)$ is vector of functions which is vector rough convex functions such that $F_*(x) \leq F(x) \leq F^*(x)$, $F_*(x)$ and $F^*(x)$ are the lower and upper approximation deterministic vectors functions of $F(x)$ respectively.

$$F_*(x) = (f_{*1}(x), f_{*2}(x), \dots, f_{*k}(x))^t, F^*(x) = (f_1^*(x), f_2^*(x), \dots, f_k^*(x))^t$$

The problem (7) can be solved in the following manner:

Firstly: Solve the following problem by using any suitable method:

$$\begin{aligned} & \min F_*(x) \\ & \text{subject to } x \in X \end{aligned}$$

If $\check{C}$ is the Pareto optimal solution set and $\check{F}_* = \{F_*(x) | x \in \check{C}\}$ is the Pareto optimal set.

Find $\check{C}_1 = \{\bar{x} \in \check{C} | F^*(\bar{x}) \in \check{F}_*\}$.

Definition (The surely optimal solution set):

If $\check{C}_1 \neq \emptyset$, then $\check{C}_1$ is called the surely optimal solution set of the problem (7) which contains all surely optimal solutions.

If $\check{C}_1 = \check{C}$, the problem has only a surely Pareto optimal solution set.

Secondly: If $\check{C}_1 = \emptyset$, solve the following problem

$$\begin{aligned} & \min F^*(x) \\ & \text{subject to } x \in X \end{aligned}$$

If $\check{C}_2$ is Pareto optimal solution set and $\check{F}^* = \{F^*(x) | x \in \check{C}_2\}$ is the Pareto optimal set.

$$\theta^* = \begin{cases} \max \left(f_1(x) + \sum_{r=1}^a \mu_r g_r(x), \dots, f_k(x) + \sum_{r=1}^a \mu_r g_r(x) \right) \\ \text{subject to } \sum_{i=1}^k \bar{w}_i \nabla f_i(x) + \sum_{r=1}^a \mu_r \nabla g_r(x) = 0 \\ \bar{w}_i > 0, \sum_{i=1}^k \bar{w}_i = 1, \mu_r \geq 0, r = 1, 2, \dots, m, \bar{w} \in S_*(\bar{x}) \cap S^*(\bar{x}) \end{cases}$$

Where $\bar{x}$ and $\bar{\bar{x}}$ are Pareto optimal solutions of primal problem on lower approximation set and on upper approximation set, respectively. $S_*(\bar{x})$ and $S^*(\bar{x})$ are stability sets of first kind problem on lower approximation set and on upper approximation set, respectively.

The dual problem can be solved in the following manner:

Firstly: Solve the following problem:

$$\begin{aligned} & \max \left(f_1(x) + \sum_{r=1}^A \mu_r G_r(x), \dots, f_k(x) + \sum_{r=1}^A \mu_r G_r(x) \right) \\ & \text{subject to } \sum_{i=1}^k \bar{w}_i \nabla f_i(x) + \sum_{r=1}^a \mu_r \nabla G_r(x) = 0 \\ & \bar{w}_i > 0, \sum_{i=1}^k \bar{w}_i = 1, \mu_r \geq 0, r = 1, 2, \dots, m, \bar{w} \in S_*(\bar{x}) \cap S^*(\bar{\bar{x}}) \end{aligned}$$

If $\dot{D}$ is Pareto optimal solution set and θ_* is Pareto optimal set.

Find $\dot{D}_1 = \{\bar{\mu} \in \dot{D} \mid \theta_*(\bar{\mu}) \in \theta^*\}$ which is a surely Pareto optimal solution set if $\dot{D}_1 \neq \emptyset$.

Secondly: If $\dot{D}_1 = \emptyset$, solve the following problem

$$\begin{aligned} & \max \left(f_1(x) + \sum_{r=1}^a \mu_r g_r(x), \dots, f_k(x) + \sum_{r=1}^a \mu_r g_r(x) \right) \\ & \text{subject to } \sum_{i=1}^k \bar{w}_i \nabla f_i(x) + \sum_{r=1}^a \mu_r \nabla g_r(x) = 0. \\ & \bar{w}_i > 0, \sum_{i=1}^k \bar{w}_i = 1, \mu_r \geq 0, r = 1, 2, \dots, m, \bar{w} \in S_*(\bar{x}) \cap S^*(\bar{\bar{x}}) \end{aligned}$$

If $\dot{D}_2$ is Pareto optimal solution set and θ_* is Pareto optimal set. The set $\dot{D} \cup \dot{D}_2$ is a possibly Pareto optimal solution set.

Definition (The surely Pareto optimal solution set):

If $\dot{C}_1 \neq \emptyset$, then $\dot{C}_1$ is called surely Pareto optimal solution set of problem (6) which contains all surely Pareto optimal solutions.

If $\dot{C}_1 = \dot{C}$; $\dot{C} \subseteq X_*$; then problem (6) has only surely Pareto optimal solutions.

Secondly: If $\dot{C}_1 = \emptyset$, $\dot{C} \subseteq X_{BN}$, solve the following problem:

$$\begin{aligned} & \min F(x) \\ & \text{subject to } x \in X_* \end{aligned}$$

If $\dot{C}_2$ is the Pareto optimal solution set of the above problem and $\dot{F}^* = \{F(x) | x \in \dot{C}_2\}$ is the Pareto optimal set of multiobjective functions.

Definition (The possibly Pareto optimal solution set):

If $\dot{C}_1 = \emptyset$, the set $\dot{C} \cup \dot{C}_2$ is called the possibly Pareto optimal solution set of problem (6) which contains all possibly Pareto optimal solutions.

The Pareto optimal set of the multiobjective functions $F(x)$ is between Pareto optimal set $\dot{F}_*$ of $F(x)$ on the upper approximation set and Pareto optimal set $\dot{F}^*$ of $F(x)$ on the lower approximation set, $\dot{F}_* \subseteq \min_{x \in X} F(x) \subseteq \dot{F}^*$.

The dual problem of the first class of MRCPP is formulated as problem (3). So the Pareto optimal set of the dual problem of the first class is between

$$\dot{\theta}_* \subseteq \max \dot{\theta}(\mu) \subseteq \dot{\theta}^*$$

where

$$\dot{\theta}_* = \begin{cases} \max \left(f_1(x) + \sum_{r=1}^A \mu_r G_r(x), \dots, f_k(x) + \sum_{r=1}^A \mu_r G_r(x) \right) \\ \text{subject to } \sum_{i=1}^k \bar{w}_i \nabla f_i(x) + \sum_{r=1}^A \mu_r \nabla G_r(x) = 0 \\ \bar{w}_i > 0, \sum_{i=1}^k \bar{w}_i = 1, \mu_r \geq 0, r = 1, 2, \dots, m, \bar{w} \in S_*(\bar{x}) \cap S^*(\bar{x}) \end{cases}$$

$$\begin{aligned} & \max \sum_{i=1}^k w_i f_i(\bar{x}) + \sum_{r=1}^m \mu_r g_r(\bar{x}) \\ & \text{subject to} \\ & \sum_{i=1}^k w_i \nabla f_i(\bar{x}) + \sum_{r=1}^m \mu_r \nabla g_r(\bar{x}) = 0, \mu_r \geq 0, r = 1, 2, \dots, m \end{aligned} \quad (5)$$

Theorem:

If $\bar{x}$ is a unique optimal solution of problem (2) then there is $\bar{\mu}_r \geq 0$ optimal solution to problem (5).

Proof:

It can be proved as the same steps as theorem 4.6 [4].

3. Duality of MRCPP

The duality of MRCPP for the two classes can be obtained by using any type of duality forms in the previous section depending on the properties of MRCPP.

3.1 Dual Problem of First Class of MRCPP:

As known the first class of MRCPP can be defined as:

$$\begin{aligned} & \min F(x) = (f_1(x), f_2(x), \dots, f_k(x))^t \\ & \text{subject to } x \in X \end{aligned} \quad (6)$$

where $F: R^n \rightarrow R^k$ is a vector of convex deterministic functions, X is the feasible region which is a convex rough set such that $X_* \subseteq X \subseteq X^*$,

$$X^* = \{x \in R^n | G_i(x) \leq 0, i = 1, 2, \dots, A\}, X_* = \{x \in X^* | g_i(x) \leq 0, i = 1, 2, \dots, a\}.$$

The problem (6) can be solved in the following manner:

Firstly: Solve the following problem by using any suitable method:

$$\begin{aligned} & \min F(x) \\ & \text{subject to } x \in X^* \end{aligned}$$

If $\hat{C}$ is the Pareto optimal solution set and $\hat{F}_* = F(\hat{C})$ is the Pareto optimal set.

Find the set $\hat{C}_1 = \hat{C} \cap X_*$.

The duality of multiobjective optimization problem has been studied by many authors see refs [6, 15, 18]. But the cone optimization is used to get the duality in many studies. Only some researchers study the duality in general form [18]. In this paper, a simple method is presented to get the duality.

The first dual form is used when $\bar{w}_i > 0, \sum_{i=1}^k \bar{w}_i = 1$ which it defined as follows:

$$\begin{aligned} \max & \left(f_1(x) + \sum_{r=1}^m \mu_r g_r(x), f_2(x) + \sum_{r=1}^m \mu_r g_r(x), \dots, f_k(x) + \sum_{r=1}^m \mu_r g_r(x) \right) \\ \text{subject to} & \end{aligned} \quad (3)$$

$$\sum_{i=1}^k \bar{w}_i \nabla f_i(x) + \sum_{r=1}^m \mu_r \nabla g_r(x) = 0$$

$$\bar{w}_i > 0, \sum_{i=1}^k \bar{w}_i = 1, \mu_r \geq 0, r = 1, 2, \dots, m, \bar{w} \in S(\bar{x})$$

Theorem:

If there is a Pareto optimal solution $\bar{x}$ such that $\bar{w}_i > 0, \sum_{i=1}^k \bar{w}_i = 1, \bar{w} \in S(\bar{x})$ where $S(\bar{x})$ the stability set of the first kind, then there is $\bar{\mu}_r \geq 0$ Pareto optimal solution to problem (3).

Proof:

It can be proved as the same steps as theorem 4.6 [4].

If problem (1) is transformed into problem (2) by using the weighted method. Then there are two dual form of problem (3), **firstly** by using Lagrange method. The theorems and concepts of convex programming problem are valid if MRCPP is transformed to convex programming problem [3] under the suitable conditions. The Lagrange function can be defined as follows:

$$\begin{aligned} \max & \theta(\mu) \\ \text{subject to} & \mu \geq 0 \end{aligned} \quad (4)$$

where $\theta(\mu) = \inf\{\sum_{i=1}^k w_i \nabla f_i(x) + \sum_{r=1}^m \mu_r \nabla g_r(\bar{x}) = 0, \mu_r \geq 0 | x \in X\}$

Secondly: The other dual form of problem (3) can be defined by differentiable properties when its Pareto optimal solution $\bar{x}$ is a unique optimal solution as follow:

The multiobjective convex programming problem (MCP) is defined as follows [4]:

$$\begin{aligned} & \min(f_1(x), f_2(x), \dots, f_k(x)) \\ & \text{subject to} \\ & X = \{x \in R^n | g_r(x) \leq 0, r = 1, 2, \dots, m\} \end{aligned} \quad (1)$$

where the functions $f_i: R^n \rightarrow R$ for $i = 1, \dots, k$ and $g_r: R^n \rightarrow R$ for $r = 1, \dots, m$ are assumed to be differentiable and convex.

If problem (1) is transformed into problem (2) by using the weighted method [4].

$$\begin{aligned} & \min \sum_{i=1}^k w_i f_i(x) \\ & \text{subject to} \\ & X = \{x \in R^n | g_r(x) \leq 0, r = 1, 2, \dots, m\} \\ & w_i \geq 0, \quad \sum_{i=1}^k w_i = 1 \end{aligned} \quad (2)$$

The whole Pareto optimal solution set is gotten if convexity assumption is presented but it fail to get the whole Pareto optimal solution set in of non-convex problems.

Theorem [4]:

Assume convexity if $x^* \in X^*$ (X^* is Pareto optimal solution set of (1)) then there exists $w \in W$ such that x^* solve problem (2).

Theorem [4]:

x^* a Pareto optimal solution of MCP if there exists $w \in W$ such that x^* solves problem (2) and if either one of the following two conditions holds:

1. $w_i > 0$ for all $i = 1, 2, \dots, k$.
2. x^* is a unique solution of problem (1).

From above relations, the stability set of the first kind of problem (1) according to w can be defined as $S(\bar{x}) = \{w \in W | \bar{x} \text{ is a Pareto optimal solution to problem (1)}\}$.

^{3*} Benha Faculty of Engineering, Benha University, Benha, Egypt.

multiobjective convex programming problem in rough environment when the multiobjective function is rough functions and the feasible region is deterministic. Also it discussed the duality when roughness is in feasible region and multiobjective is deterministic. Section 4 includes the conclusions.

2. Basic Concepts

RST is a new mathematical theory introduced by Pawlak in the early 1980s to deal with vagueness or uncertainty, [10,11,12,13,14]. It is a very rich area for research, [1,16, 17]. RST expresses vagueness by employing a boundary region of a set and not by means of membership function, [2,7,12,19]. The basic concept of RST is the approximation of indefinable sets via definable sets or ordinary sets, [1,2,10,12]. The rough function is a new concept based on the RST [7,12,19].

Definition (Convex Rough Set) [19]:

A rough set is convex if its lower and upper approximation sets are convex.

A rough function f is a function without explicit formula but bounded by several explicit functions because it is not known precisely. It is given as result of modelling the problem due to imperfect data. For simplification on our study the rough function will be defined by two functions.

Definition (Rough Function):

A function $f(x)$ is a rough function if it is not known precisely but it is only known that the function values of $f(x)$ are bounded by two deterministic functions $f_*(x), f^*(x)$ such that $f_*(x) \leq f(x) \leq f^*(x)$ at any $x \in X$ where $f_*(x), f^*(x)$ are lower and upper approximation function respectively. The surely values of $f(x)$ are the values of $f_*(x), f^*(x)$ at $x \in X$ such that $f_*(x) = f^*(x)$.

Definition (Convex Rough Function):

A rough function $f(x)$ is called convex rough function on the set X if its lower and upper approximation functions are convex.

On duality of Multiobjective Rough Convex Programming Problems

M.S.A.Osman¹, M.H.Eid², M.A.Elsisy^{3*}

Abstract:

Duality assertions are very important in optimization from the theoretical as well as from the numerical point of view. So this paper presents duality of multiobjective rough convex programming problems in rough environment when the multiobjective function is deterministic and roughness is in feasible region. Also it discussed the duality when roughness in multiobjective function and the feasible region is deterministic. The concepts and some theorems of duality in the rough environment are discussed. Also, the procedure of solution of these kind of problems described.

Key words: Rough set; rough function; multiobjective rough convex programming problem; duality.

I. Introduction

The mathematical programming in rough environment was introduced in [2,5,7,19]. The multiobjective rough convex programming problem (MRCPP) can be classified according to existence of roughness is in multiobjective function or constraints. The mathematical problems can be classified into three classes, [7]. First class: MRCPP with rough feasible set and deterministic multiobjective function. Second class: problems with deterministic feasible set and rough multiobjective function. Third Class: problems with rough feasible set and rough multiobjective function. There are two solution sets of the MRCPP; surely Pareto optimal solution set and possibly Pareto optimal solution set. The aim of this paper is to present the concepts of duality of MRCPP when roughness is in the multiobjective function and the feasible region is deterministic. Also it discussed the duality when roughness is in feasible region and multiobjective is deterministic.

This paper consists of four sections. Section 2 introduces the mathematical formulation of MRCPP and presents basic concepts of rough set theory. Section 3 presents duality theorems of

¹Al Asher University, Al Asher of Ramadan, Egypt.

²Shoubra Faculty of Engineering, Benha University, Benha, Egypt.

[14] Gang Li, Fuzzy goal programming – A parametric approach, Information Sciences 195 (2012) 287–295.

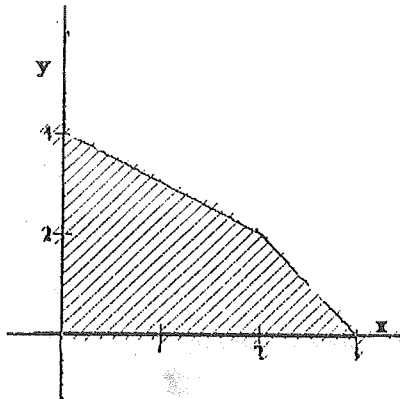
[15] E. Dopazo, M. Ruiz-Tagle, A parametric GP model dealing with incomplete information for group decision-making, Applied Mathematics and Computation 218 (2011) 514–519 .

[16] Ching-Ter Chang, Binary fuzzy goal programming, European Journal of Operational Research 180 (2007) 29–37.

[17] Chin-Nung Liao, formulating the multi-segment goal programming, Computer and Industrial Engineering 56(2009) 138-141.

- [2] V. Chankong, V.V. Haimes" Multiobjective Decision Making: Theory and Methodology", North-Holland, Amsterdam, 1983.
- [3]Jerald P. Dauer, Robert J. Krueger,An iterative approach to goal programming, Operational Research Quarterly(1970-1977),Vol. 28.No3 Part 2 (1977) ,671-681.
- [4] R.P. Hamalainen, J. Mantysaari, Dynamic multi-objective heating optimization, European Journal of Operational Research 142 (2002) 1–15.
- [5] L.A. Zadeh, Fuzzy sets, Information and Control 8 (1965) 338–353.
- [6] T.L. Saaty, Exploring the interface between hierarchies multi-objectives and fuzzy sets, Fuzzy Sets and Systems 1 (1978) 57–68.
- [7] R.P. Hamalainen, J. Mantysaari, Dynamic multi-objective heating optimization, European Journal of Operational Research 142 (2002) 1–15.
- [8] Ignizio, J. P. (1976). Goal programming and extensions. Lexington Series. Lexington, MA: D. C. Heath & Company.
- [9] Carlos Romero, A general structure of achievement function for a goal programming model, European Journal of Operational Research 153 (2004) 675–686.
- [10] R. Narasimhan, Goal programming in a fuzzy environment, Decisions and Sciences 11 (1980) 325–336.
- [11] J.R. Rao, R.N. Tiwari, B.K. Mohanty, A preference structure on aspiration levels in a goal programming problem—a fuzzy approach, Fuzzy Sets and Systems 25 (1988) 175–182.
- [12] S.S. Rao, K. Sundararaju, B.G. Prakash, C. Balakrishna, Fuzzy goal programming approach for structural optimization, AIAA Journal 30 (5) (1992) 1425–1432.
- [13] P.A. Rubin, R. Narasimhan, Fuzzy goal programming with nested priorities, Fuzzy Sets and Systems 14 (1984) 115–129.

$$x, y, d_1^-, d_1^+, d_2^-, d_2^+ \geq 0, \alpha_1, \alpha_2 \in [0,1] \quad (10)$$



At the point (2,2) we get $\alpha_1 = 0.3, \alpha_2 = 0.4$.

At the point (0,0) we get $\alpha_1 = 0.58, \alpha_2 = 0.7$

$$\therefore \alpha_1 \in [0.3, 0.58], \alpha_2 \in [0.4, 0.7]$$

$$\therefore T = \{(\alpha_1, \alpha_2) : \alpha_1 \in [0.3, 0.58], \alpha_2 \in [0.4, 0.7]\}$$

Remark:

The previous example is treated graphically since the problem is represented in two dimensions for the variables x, y however the simplex method can be used to handle this problem in general case due to the use of the iterative approach for treating such problems.

5. Conclusion

In this article the achievement stability set for parametric linear goal programming problem when Parameters exist in the objective function is introduced by using the iterative approach and an illustrative example is introduced.

References:

- [1] A. Charnes, W.W. Cooper Management Models and Industrial Applications of Linear Programming, vols. 1&2, John Wiley and Sons, New York, 1961.

Subject to

$$3x + 2y + d_1^- - d_1^+ = 25 ,$$

$$x + y \leq 4 ,$$

$$2x + y \leq 6 ,$$

$$x, y, d_1^-, d_1^+ \geq 0 \quad , \quad \alpha_1 \in [0,1] \quad (7)$$

Let $\alpha = 0.3$, then the optimum of the problem is $x = 2, y = 2$,

$$d_1^- = 15, d_1^+ = 0, Z_1^* = 10.5.$$

Second we will solve:

$$\text{Min} Z_2 = \alpha_2 d_2^+ + (1 - \alpha_2) d_2^-$$

Subject to

$$3x + 2y + d_1^- - d_1^+ = 25 ,$$

$$5x + 3y + d_2^- - d_2^+ = 32 ,$$

$$x + y \leq 4 ,$$

$$2x + y \leq 6 ,$$

$$\alpha_1 d_1^+ + (1 - \alpha_1) d_1^- = 10.5 ,$$

$$x, y, d_1^-, d_1^+, d_2^-, d_2^+ \geq 0 \quad , \quad \alpha, \alpha_2 \in [0,1] \quad (9)$$

Let $\alpha = 0.3, \alpha_2 = 0.4$.Then we get $x = 2, y = 2, d_1^- = 15, d_1^+ = 0$,

$d_2^- = 16, d_2^+ = 0$ And $Z_2^* = 9.6$.And to make Z_2^* be stable we will solve:

$$\text{Min} Z_2 = \alpha_2 d_2^+ + (1 - \alpha_2) d_2^-$$

Subject to

$$3x + 2y + d_1^- - d_1^+ = 25 ,$$

$$5x + 3y + d_2^- - d_2^+ = 32 ,$$

$$x + y \leq 4 ,$$

$$2x + y \leq 6 ,$$

$$\alpha_1 d_1^+ + (1 - \alpha_1) d_1^- = 10.5 ,$$

$$\alpha_2 d_2^+ + (1 - \alpha_2) d_2^- = 9.6 ,$$

$$d_j^-, d_j^+, X \geq 0, \quad j = 1, \dots, i \quad \alpha_i \in [0,1] \quad \forall i \quad (5)$$

Where Z_j^* is the achievement value of the goal j , $G_i(X)$ are linear functions

Definition (4): The achievement stability set:

$$T = \{(\alpha_1, \alpha_2, \dots, \alpha_m) \in R^m : \text{Min } Z_i = \alpha_i d_i^+ + (1 - \alpha_i) d_i^- = Z_i^*, Z_i^* \text{ is constant, } i = \{1, 2, \dots, m\}\}$$

is the achievement stability set where it is the values of α_i which make the achievement value of all goals to be constant.

4-Solution algorithm:

step1: Solve the problem P_1 at a certain value of $\alpha_1 \in [0,1]$ and find Z_1^*

step2: Set $i = 2$

Step3: Solve the attainment problem P_i at a certain value of $\alpha_i \in [0,1]$ and find Z_i^* .

Step4: Set $i = i + 1$ and go to step3.

Step5: If $i = m$, go to step6.

Step6: find the values of α_i which make Z_i^* be constant.

Step7: stop.

Example (1): Consider the following problem PGPP: $\text{Min} Z =$

$$P_1[\alpha_1 d_1^+ + (1 - \alpha_1) d_1^-] + P_2[\alpha_2 d_2^+ + (1 - \alpha_2) d_2^-]$$

Subject to

$$3x + 2y + d_1^- - d_1^+ = 25,$$

$$5x + 3y + d_2^- - d_2^+ = 32,$$

$$x + y \leq 4,$$

$$2x + y \leq 6,$$

$$x, y, d_i^-, d_i^+, \geq 0 \quad , \quad i = 1, 2, \quad \alpha_1, \alpha_2 \in [0,1] \quad (6)$$

First we will solve:

$$\text{Min} Z_1 = \alpha_1 d_1^+ + (1 - \alpha_1) d_1^-$$

$$G_1(X) + d_1^- - d_1^+ = b_1 , \quad (3)$$

$$X \in S \subset R^n$$

$$d_1^-, d_1^+, X \geq 0 , \quad \alpha_1 \in [0,1]$$

If Z_1^* denotes the optimum value of this problem, then we say that Z_1^* is the degree of the attainment for goal 1.

Now the attainment problem for goal 2 is equivalent to problem P_2 :

$$P_2: \text{ minimize } Z_2 = \alpha_2 d_2^+ + (1 - \alpha_2) d_2^-$$

Subject to:

$$G_1(X) + d_1^- - d_1^+ = b_1 ,$$

$$G_2(X) + d_2^- - d_2^+ = b_2 , \quad (4.a)$$

$$\alpha_1 d_1^+ + (1 - \alpha_1) d_1^- = Z_1^* \quad (4.b)$$

$$X \in S \subset R^n$$

$$d_1^-, d_1^+, X \geq 0 , \quad \alpha_2 \in [0,1] \quad (4)$$

Notice that the constraints (4.a), (4.b) imply that in trying to attain goal 2 we are not sacrifice our previously determined the degree of attainment of goal 1. Letting Z_2^* denote the solution of this problem. We can then proceed to goal 3 and so on. We can now write the general attainment problem as follows:

Definition (3): The general attainment problem P_i for goal :

The general attainment problem P_i for goal i can be defined as[3]:

$$P_i: \text{ minimize } Z_i = \alpha_i d_i^+ + (1 - \alpha_i) d_i^- \quad i \in \{2, \dots, m\}$$

Subject to:

$$G_j(X) + d_j^- - d_j^+ = b_j , \quad j = 1, \dots, i$$

$$\alpha_j d_j^+ + (1 - \alpha_j) d_j^- = Z_j^*, \quad j = 1, \dots, i - 1$$

$$X \in S \subset R^n$$

denoted by d_i^-) or over achievement (positive deviation and denoted by d_i^+) of goal i . $S, G_i(X) \ i = 1, \dots, m$ are the system constraints and goal constraints of the problem respectively, w_i^-, w_i^+ are the weights.

Notice that listing of goals in (1) corresponds to the order in which we wish to achieve these goals. In other words our main objective is to attain the first goal $G_1(X)$ if possible subject to the system constraints $X \in S$ and the goal constraint corresponding to this goal. If this not possible we wish to have $G_1(X)$ be as close to satisfying this goal as it possible. We then get the values of corresponding parameter which make the first goal stable we then satisfy the second goal $G_2(X)$ as close as possible subject to the system constraints $X \in S$ and its goal constraint and subject to the additional constraint that we preserve our previous degree of attainment of goal 1. we then get the values of the corresponding parameter which make $G_2(X)$ is stable. Repeat the previous steps until $i = m$. A key element of GP model is the achievement function that represents a mathematical expression of the unwanted deviation variables [9].

Definition (1): The achievement function:

The goal programming achievement function is an ordered vector of a dimension equal to the number (i) of preemptive priorities within the problem and expresses the level of achievement of each objective set within a given priority [8].

3- Parametric goal programming problem when parameters exist in the objective function.

Definition (2): The attainment problem for goal 1:

The attainment problem for goal 1 to be the optimization problem P_1 :

$$P_1: \text{minimize } Z_1 = \alpha_1 d_1^+ + (1 - \alpha_1) d_1^-$$

Subject to:

To incorporate uncertainty and imprecision into the formulation, the fuzzy set theory, initially proposed by Zadeh in 1965 [6] is introduced in the field of conventional MODM problems, where aspiration level of objectives are assigned in an imprecise manner. Introducing fuzzy uncertainty and imprecision into the goal programming problems, Narasimhan [10] initially presented fuzzy goal programming (FGP) by using membership functions, and some relative work was done [11–13]. Gang Li [14] introduced a model that deals with the sum of weighted negative deviations between the desirable achievement degree and the common target. E. Dopazo, M. Ruiz-Tagle[15] have been established an optimization problem in which a similarity function and a parametric compromise function are defined. Associated to this problem, a logarithmic goal programming formulation is considered to provide an effective procedure to compute the solution. Ching-Ter Chang[16] proposed a new idea of how to program the binary fuzzy goal programming model. The binary fuzzy goal programming model can then be solved using the integer programming method.

Assume that, the goal programming in problem consists of m goals and S system constraints, X decision variables; the problem formulation can be defined as follows:

$$\min Z(d^-, d^+) = \sum_{i=1}^m P_i (w_i^- d_i^- + w_i^+ d_i^+).$$

Subject to

$$G_i(X) + d_i^- - d_i^+ = b_i, \quad i = 1, \dots, m \quad (1)$$

$$X \in S$$

$$d_i^-, d_i^+, X \geq 0, w_i^-, w_i^+ \geq 0$$

The objective function seeks to minimize the deviations from the preferred aspiration levels. $X = (x_1, x_2, \dots, x_n)$ is the decision vector. A deviation variable reflects the under achievement (negative deviation and

So that GP is a good tool to enable the decision maker to realize the balance between the available resources and the desired goals. The objective function of GPP focused on minimizing the deviation between the available resources and the required goals. GP is one of the main approaches for solving multicriteria decision making problems. This paper is organized as follows: Section2, the basic concepts of goal programming problems was introduced.

Section3, parametric goal programming (PGPP) when parameters in the objective function are presented. In Section 4, solution algorithm and example are given. Section 5, includes conclusion and recommendations for further studies.

2. The Goal Programming Problem:

Goal programming is considered as one of the three main approaches for solving multicriteria decision making problem, the other two approaches are vector optimization problem (VOP) and the Interactive approach. In GP decision makers must initially give priorities for their goals .The GP gives an optimal solution. To solve the goal programming problem we can use the modified simplex method which is an algebraic procedure. It was developed by Charnes and Cooper [1]. And we can use the iterative approach method which is a very powerful method for solving the linear, nonlinear, integer and other goal programming problems using the corresponding optimization algorithm in an Iterative analysis equivalent to that available for the corresponding optimization technique[3]. The GP approach of multi-criteria problems has received increasing interest due to its modeling flexibility and conceptual simplicity [4]. In GP problem, it is practical to consider that there are different importance and priorities of the goals, and a lot of research is done with this problem. The priority structure is assigned by using different methods, such as fuzzy analytical hierarchy process [5].

On the achievement stability set for parametric linear goal programming problem.

Osman .M.S¹, Maaty .M. A² and Farahat. F.A³

Abstract:

Goal Programming (GP) has many significant advances in recent years, particularly in the area of intelligent modeling and solution analysis as it is powerful and effective methodology for the modeling, solution, and analysis of problems having multiple and conflicting objectives. In this paper, the achievement stability set for parametric linear goal programming problem will be investigated when parameters exist in the objective function, finally an illustrative example will be introduced.

Keywords: Goal programming, iterative approach, achievement function, parametric analysis.

1. Introduction:

Goal programming (GP) is a tool that has been proposed as a model and approach for the analysis of problems involving multiple, conflicting objectives with different measures units. Goal programming (GP), firstly introduced by Charnes and Cooper in the early 1960's [1]. This technique has been applied in different areas to support the decision maker such as multiobjective decision making (MODM) problem [2]. The GP approach of multi-criteria problems has received increasing interest due to its modeling flexibility and conceptual simplicity[7]. For solving goal programming problem (GPP), it is necessary to establish a hierarchy of importance among these goals so that lower priority goals are considered only after higher priority ones are satisfied. Since it is not always possible to achieve every goal according to the decision maker desires, GP attempts to reach a satisfactory solution.

¹ Alasher university 10th of Ramadan city

^{2,3} Higher technological institute 10th of Ramadan city

- [24] De Castro, L.N. and J. Timmis. An artificial immune network for multimodal function optimization. in Proc. of IEEE congress on evolutionary computation 2002;1:699-674.
- [25] Orhan Engin, Alper Döyen. A new approach to solve hybrid flow shop scheduling problems by artificial immune system, Future Generation Computer Systems 2004;20:1083-95.
- [26] Tsai, J. T., Ho, W. H., Liu, T. K., & Chou, J. H. Improved immune algorithm for global numerical optimization and job shop scheduling problems. Applied Mathematics and Computation 2007;194:406-424.
- [27] El-Sherbiny, Mahmoud M, Ibrahim Yasser M. An artificial immune algorithm with alternative mutation methods: applied to the student project assignment problem, International Conference on Innovation and Information Management (ICIIM2012), Chengdu, China, January 7-8, 2012.
- [28] Altassan, Khaled. M, El-Sherbiny, Mahmoud M, Abid, Ahmed D. Solving fixed charge transportation problems using improved artificial immune system, 1st International Conference on Decision Modeling (ICDeM2012), Sintok, Kedah, Malaysia, March 13-16, 2012;101-108.
- [29] Ada, G.L. & Nossal, G. The clonal selection theory, Sci. Am. 1987;2:50-57.
- [30] Chien-Wei Chu, Min-Der Lin, Gee-Fon Liu, Yung-Hsing Sung. Application of immune algorithms on solving minimum-cost problem of water distribution network. Mathematical and Computer Modeling, 2008;48:1888-1900.
- [31] Chen, T. & You, P.S. Immune algorithms-based approach for redundant reliability problems with multiple component choices, Comput. Ind. Eng 2005;56(2):195-205.
- [32] Jerne, N. K. Towards a network theory of the immune system. Annu Immunol,1974;125:373-89.
- [33] Taguchi, G. Introduction to quality engineering. White Plains: Asian Productivity Organization/ UNIPUB. 1986.
- [34] Phadke, M.S. Quality engineering using robust design. Prentice-Hall, NJ, 1989.
- [35] Yunus, M., Rahman, F. Optimization of usage parameters of ceramic coatings in high temperatures applications using Taguchi design, Int. Journal of Engineering Science and Technology (IJEST) 2011;8: 6364-71.
- [36] Zalinda Othman, Mohammad-Reza Rostamian Delavar, Sarah Behnam, Sina Lessanibahri. Adaptive genetic algorithm for fixed-charge transportation problem, International Multi Conference of Engineers & Computer Sciences (IMECS) March 16-18, 2011.

- [6] Balinski, M.L. Fixed cost transportation problem. *Naval Research Logistics Quarterly* 1961;8: 41-54.
- [7] Sun, M., Aronson, J.E. Mckeown, P.G. A tabu search heuristic procedure for the fixed charge transportation problem, *European Journal of Operational Research*, 1998;106:441-456.
- [8] Gen, M., K. Ida, and Y.Z. Li. Bicriteria transportation problem by hybrid genetic algorithm. *Computers and Industrial Engineering*, 1998;35(12):363-366.
- [9] Hajiaghahi, M. Keshteli, S. Molla-Alizadeh-Zavardehi, R. Tavakkoli-Moghaddam. Addressing a nonlinear fixed-charge transportation problem using a spanning tree-based genetic algorithm, *Computers & Industrial Engineering* 2010;59:259-271.
- [10] Kannan, G., Sasi Kumar, P., & Vinay, V. P. Comments on "Nonlinear fixed charge transportation problem by spanning tree-based genetic algorithm" by Jung-Bok Jo, Yinzhen Li, Mitsuo Gen, *Computers & Industrial Engineering* (2007). *Comput Ind Eng* 2008;55:533-34.
- [11] Gottlieb, J., B.A. Julstrom, F.Rothlauf, and Raidl, G.R.: Prüfer numbers: A poor representation of spanning trees for evolutionary search. In: Spector L, et al., eds. *Proc. of the 2001 Genetic and Evolutionary Computation Conf.* San Francisco: Morgan Kaufmann Publishers 2001:343-50.
- [12] Raidl, B. A., Julstrom, G.R. Edge-sets: An effective evolutionary coding of spanning trees. *IEEE Transactions on Evolutionary Computation* 2003;7(3):225-239.
- [13] Su, S., Zhan, D.C. New genetic algorithm for the fixed charge transportation problem, *Proceedings of the 6th World Congress on Intelligent Control and Automation* 2006:7039-43.
- [14] El-Sherbiny, M. Mahmoud., Alhamali, M. Rachid. A hybrid particle swarm method with artificial immune learning for solving the fixed charge transportation problem. *Comput Ind Eng* 2012 submitted for publication.
- [15] Altassan, Khaled M, El-Sherbiny M. M., Ibrahim Yasser M, Abid Ahmed D. A novel artificial immune algorithm for solving fixed charge transportation problems, *International Conference on Innovation and Information Management (ICIIM2012)*, Chengdu, China, January 7-8, 2012.
- [16] Molla-Alizadeh-Zavardehi, S., Hajiaghahi-Keshteli, M., & Tavakkoli-Moghaddam, R. Solving a capacitated fixed-charge transportation problem by artificial immune and genetic algorithms with a Prüfer number representation. *Expert Systems with Applications* 2011;38:10462-74.
- [17] El-Sherbiny, M. M., (2012). Comments on "Solving a capacitated fixed-charge transportation problem by artificial immune and genetic algorithms with a Prüfer number representation. *Expert Systems with Applications*, 38, (2011)10462-10474." *Expert Systems with Applications* 2012;39:11321-22.
- [18] Kowalski, K and Lev, Benjamin. On step fixed-charge transportation problem, *OMEGA, The International Journal of Management SCI.* 2008; 36(5):913-917.
- [19] El-Sherbiny, M. M. Alternate mutation based artificial immune algorithm for step fixed charge transportation problem *Egyptian Informatics Journal.* 2012;13,123-134.
- [20] Altassan, Khaled. M., El-Sherbiny, M. M., Sasidhar, Bokkasam. Handling the step fixed charge transportation problem, *1st International Conference on Decision Modeling (ICDeM2012)*, Sintok, Kedah, Malaysia, March 13-16, 2012:109-116.
- [21] Timmis, J., T Knight, L. N. Catro and Hart. An overview of artificial immune systems. *Computation in cells and tissues: perspectives and tools thought. Natural Computation Series*, Springer-Verlag, 2004:51-86.
- [22] De Castro, L. N., & Timmis, J. *Artificial immune system: A new computational intelligence approach.* New York: Springer; 2002.
- [23] De Castro, L.N. & Timmis, J. *Artificial immune systems as a novel soft computing paradigm*, *Soft Computing. J.* 2003;7 (8):526-544.

Table 10: Levels that give the best values

Factor	B	C	D	E	Best Level
A	(3,4)	(3,2)	(3,1)	(3,13)	3
B		(4,2)	(4,1)	(4,13)	4
C			(2,1)	(2,13)	2
D				(4,13)	1
Best Level	4	2	1	13	

4. Conclusion

This paper has proposed a new version of FCTP in a form of step capacitated two stage supply chain network (SC2SCN) in which a given number of intermediate distribution centers (DCs) are located between the supply plant and the customer locations. In this problem the DCs are considered to be capacitated in order to supply demands of each customer and also the opening cost for each potential DC and each route are considered. In addition, that opening cost for each DC and each route are considered to be dependent on the amount shipped. Thus the cost structure causes the value of the objective function to behave like a step function. The description and the mathematical model of such super NP-hard problem are presented. In order to solve such problem, this paper proposes an artificial immune algorithm referred to as AI2SCN. The proposed AI2SCN algorithm, guarantees the feasibility of all the generated antibodies. Due to the significant role of AI2SCN's factors on the performance quality, relative percentage deviation (*RPD*) and signal noise ratio (*S/N ratio*) methods are applied in order to emphasize the factors with the highest impact, calibrate and find the best setup for the AI2SCN algorithm.

Future work includes Investigating using other metaheuristic techniques combined with the proposed decoding and shipping algorithms for solving other problems such as capacitated FCTP, Multi-Step FCTP. In addition, it is proposed to carry out further experimentation with parameters of the AI2SCN algorithm and testing it on other real life problems.

References

- [1] Jawahar, N., Balaji, A.N. A genetic algorithm for the two-stage supply chain distribution problem associated with a fixed charge. *European Journal of Operational Research* 2009;194:496-537.
- [2] Hitchcock, F. L. Distribution of a product from several sources to numerous localities. *The Journal of Mathematics and Physics*. 1941;20:224-230.
- [3] Diaby, M. Successive linear approximation procedure for generalized fixed-charge transportation problems. *Journal of Operational Research Society*. 1991;42(11):991-1001.
- [4] Adlakha V, Kowalski K. On the fixed-charge transportation problem, *OMEGA* 1999;27: 381-8.
- [5] Sun, M., McKeown, P.G. Tabu search applied to the general fixed charge problem, *Annals of Operations Research* 1993;41(14):405-420.

Table 8: Response Table for Mean *RPD*

Level	A	B	C	D	E
1	6.82	7.31	5.97	5.78	8.61
2	7.20	7.49	6.83	7.65	6.80
3	7.58	7.21	7.48	7.00	11.87
4	6.61	6.20	7.93	7.79	8.26
5					6.93
6					3.75
7					6.63
8					6.64
9					6.59
10					8.72
11					8.96
12					8.53
13					1.63
14					5.36
15					8.08
16					5.48
Delta	0.96	1.28	1.96	2.01	10.24
Rank	5	4	3	2	1

Table 9: Response Table for Mean S/N ratio

Level	A	B	C	D	E
1	-88.72	-88.76	-88.65	-88.63	-88.9
2	-88.75	-88.78	-88.72	-88.79	-88.7
3	-88.78	-88.75	-88.78	-88.74	-89.1
4	-88.71	-88.67	-88.81	-88.8	-88.8
5					-88.7
6					-88.5
7					-88.7
8					-88.7
9					-88.7
10					-88.9
11					-88.9
12					-88.9
13					-88.3
14					-88.6
15					-88.8
16					-88.6
Delta	0.08	0.1	0.16	0.17	0.83
Rank	5	4	3	2	1

provides the optimal interaction combinations for each pair of factors, based on the levels which provide the best performance (minimum values) of mean *RPD*. It can be observed that the best combination of levels for the factor A and C which provide the minimum mean *RPD* is 3, 2. Further, it is observed that the optimal levels for the other three factors B, D and E remain the same (4,1, and 13 respectively) as those obtained through the analysis based on the mean effects. Thus, by considering the interaction effects as well, the optimal levels for the factors A to E for the proposed AI2SCN algorithm will be 3,4,2,1 and 13 respectively.

Therefore, two different setup for the proposed AI2SCN algorithm are extracted based the mean affect which is 4,4,1,1 and 13 and the other based on the interaction effects which are 3,4,2,1 and 13 respectively. In order to further confirm which setup is the best, either the setup based on main effects alone which is 4,4,1,1,13 or the setup considering the interaction effects, which is 3,4,2,1,13. Therefore optimum setting based on the main effect is 50 antibodies as a population size, 30 as the number of clones, threshold ratio at 5%, applying the first affinity function (16) and applying the Non-uniformity Swap MF. Alternatively, the optimum setting based on the interaction effect is 45 antibodies as a population size, 30 as the number of clones, threshold ratio at 10%, applying the first affinity function (16) and applying the Non-uniformity Swap MF.

The results for the test problems in Table 6 using both the setups are evaluated and compared using paired t-tests in order to confirm the best setup for the proposed AI2SCN algorithm. The results of t-tests are presented in Table 10. It was found that the t-value is -1.46 and the corresponding p-value is 0.156. Therefore, it can be observed that there is no significant difference between the two setups. Hence, both setups can be used alternatively for the AI2SCN proposed algorithm. Also, it can be concluded that both main effect and interaction effect can be used to extract the best setup of any algorithm.

increase in the threshold ratio. Regarding the clone number (B), the fourth level with 30 particles provides the best performance compared to other levels. Further, the level of the best performing population size (A) is 50, which is assigned level 4. As can be seen, the S/N ratios illustrate the best parameters for the factors, which confirm the same results as the *RPD* values.

Therefore, based on the mean effects depicted in Figs. 7 and 8, as mentioned above, the rank of the factors based on their impact are the mutation function (E), the affinity function (D), removable threshold ratio (C), the clone number (B), and the population size (A), respectively. Traditionally the optimal setting is defined by taking the best level of each factor for any newly proposed algorithm [16, 35, 36]. Thus the best levels for the factors A to E for the proposed AI2SCN algorithm will be 4,4,1,1 and 13 respectively.

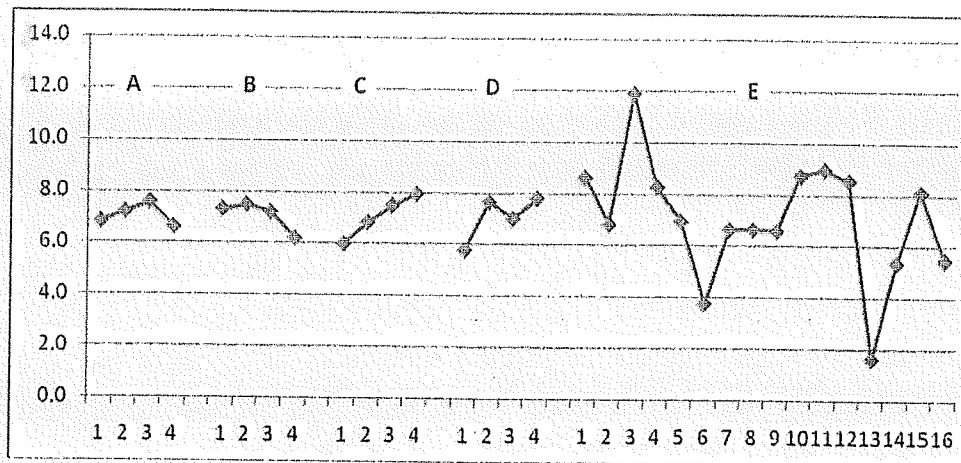


Fig. 7. Mean RPD plot for each level of the factors in AI2SCN.

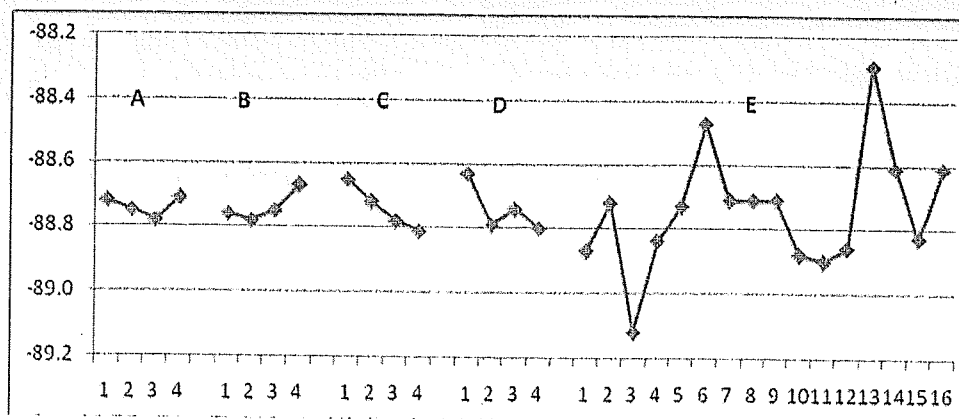


Fig. 8. Mean S/N ratio plot for each level of the factors in AI2SCN.

However, such an approach will ignore the effect of interaction among the factors. The best settings based on the mean effects alone may not always provide the optimal setting globally. Hence, the proposed optimal setting considers the effect of interactions between the factors as well. Table 8

columns indicate the actual levels of these factors. The factorial experiments are designed and analysed using MINITAB 14.1 statistical software.

Table 7: The modified orthogonal array L64 for the proposed AI2SCN algorithm

Trial	A	B	C	D	E	Trial	A	B	C	D	E	Trial	A	B	C	D	E	Trial	A	B	C	D	E
1	1	1	1	1	1	17	1	1	1	1	5	33	1	1	1	1	9	49	1	1	1	1	13
2	1	2	2	2	2	18	1	2	2	2	6	34	1	2	2	2	10	50	1	2	2	2	14
3	1	3	3	3	3	19	1	3	3	3	7	35	1	3	3	3	11	51	1	3	3	3	15
4	1	4	4	4	4	20	1	4	4	4	8	36	1	4	4	4	12	52	1	4	4	4	16
5	2	1	2	3	4	21	2	1	2	3	8	37	2	1	2	3	12	53	2	1	2	3	16
6	2	2	1	4	3	22	2	2	1	4	7	38	2	2	1	4	11	54	2	2	1	4	15
7	2	3	4	1	2	23	2	3	4	1	6	39	2	3	4	1	10	55	2	3	4	1	14
8	2	4	3	2	1	24	2	4	3	2	5	40	2	4	3	2	9	56	2	4	3	2	13
9	3	1	3	4	2	25	3	1	3	4	6	41	3	1	3	4	10	57	3	1	3	4	14
10	3	2	4	3	1	26	3	2	4	3	5	42	3	2	4	3	9	58	3	2	4	3	13
11	3	3	1	2	4	27	3	3	1	2	8	43	3	3	1	2	12	59	3	3	1	2	16
12	3	4	2	1	3	28	3	4	2	1	7	44	3	4	2	1	11	60	3	4	2	1	15
13	4	1	4	2	3	29	4	1	4	2	7	45	4	1	4	2	11	61	4	1	4	2	15
14	4	2	3	1	4	30	4	2	3	1	8	46	4	2	3	1	12	62	4	2	3	1	16
15	4	3	2	4	1	31	4	3	2	4	5	47	4	3	2	4	9	63	4	3	2	4	13
16	4	4	1	3	2	32	4	4	1	3	6	48	4	4	1	3	10	64	4	4	1	3	14

5.2. Analysis of the Factorial Experiments

The orthogonal array for the factorial experiment using 64 trials is presented in Table 7. The mean *RPD* values of the trials are averaged at each level, as presented in Table 8. Similarly the S/N ratios are averaged at each level and are presented in Table 9. These values are plotted in Figs.7 and 8 respectively. The response tables show the average of each response characteristic (mean *RPD*) for each level of each factor. The table includes ranks based on Delta statistics, which compare the relative magnitude of effects. The ranks based on Delta values; rank 1 to the highest Delta value, rank 2 to the second highest, and so on are presented in Table 9. The ranks indicate the relative importance of each factor to the response. It can be observed that the order of the relative impact rankings on the performance of the proposed AI2SCN algorithm are the mutation function (E), the affinity function (D), removable threshold ratio (C), the clone number (B), and the population size (A), respectively. This means the mutation function has the highest impact on the performance of the proposed AI2SCN algorithm followed by the other factors.

Regarding the mutation function (E), the thirteenth mutation function followed by the sixth mutation function provide the best performance than the other mutation functions. Regarding the affinity function (D), the first affinity function (17) followed by the third affinity function (19) considered at levels 1 and 3 provide the best performance than the other two affinity functions, considered as levels 2 and 4 respectively. Similarly, for the removable threshold ratio (C), the ratio at level 1 (5%) provides the best performance among the other ratios. It is noted that the performance deteriorates with

the best calibration of the proposed algorithm. The characteristics of these problems are presented in Table 6.

Table 6: Characteristics of SC2SCN test problems

Problem size	Range of supply and demand		Rang of variable costs		Rang of fixed costs($f_{ij,1}$)		Rang of fixed costs($f_{ij,2}$)	
	Lower Limit	Upper Limit	Lower Limit	Upper Limit	Lower Limit	Upper Limit	Lower Limit	Upper Limit
5x10	200	500	1	9	10	50	30	90
10x10	300	500	1	9	100	200	200	400
10x15	500	1000	1	9	100	500	200	600
15x15	500	2000	1	9	100	500	200	600
15x20	1000	3000	1	9	100	500	200	700
20x20	1000	3000	1	9	100	500	200	700

5.1. Factorial Design

A full factorial design, which tests all the possible combinations of the factors, is the widely-used method especially when the number of combinations (instances) of the factors is relatively small [11, 12]. In our case, the total number of instances of the five factors considered is $4 \times 4 \times 4 \times 4 \times 16 = 16384$ instances. Considering that each experiment, instance in this context, is run for 5 times and we have 6 problems to consider, the total number of runs required for a full factorial design is $16384 \times 5 \times 6 = 491520$ runs. This number of runs considers being extremely large. When the required number of experiments to find an optimal combination of parameters is very large, several experimental designs have been suggested to reduce the number of experiments that are needed [32]. Among several experimental design techniques, the Taguchi method has been successfully applied for a systematic approach for optimization problems [33, 34]. The performance on different combinations of parameters values is tested according to the Taguchi design of experiments (DOE). Accordingly, a cost-efficient approach, popularized by Taguchi in the mid-1980s, is applied [33]. The Taguchi method uses an orthogonal array to reduce the total number of runs required. In AI2SCN, three degrees of freedom for each factor with four levels ($3 \times 4 = 12$) and 15 degrees of freedom for the only factor with 16 levels. Thus, the sum of the required degrees of freedom is $1 + 3 \times 4 + 15 = 28$. Therefore, the appropriate array must have at least 28 rows. L64 ($4^4, 16^1$) is an appropriate array that satisfies these conditions and factor with 16 levels is adjusted and includes 64 trials instead of 16384. Table 7 shows the modified orthogonal array L64, where control factors are assigned to the columns of the orthogonal array and the corresponding integers in these

		$D(2) = AF_1 = \frac{1}{v(P_d(t))}$ $D(3) = AF_2 = \frac{v(P_d(t)) - v(P_g(t))}{v(P_d(t))}$ $D(4) = AF_3 = \frac{v(P_d(t)) - v(P_g(t))}{v(P_g(t))}$
Mutation function	E	E(1) = 2PointSwap E(2) = Inverse-Swap E(3) = Neighbor-Swap E(4) = Uniform-2PointSwap E(5) = Random-2PointSwap E(6) = RangUniformSwap E(7) = Time-2PointSwap E(8) = Rand-Inverse Swap E(9) = Time-Inverse Swap E(10) = Rand-Neighbor Swap E(11) = Time-Neighbor Swap E(12) = Normalized-Fitness-Swap E(13) = Non-uniformity Swap E(14) = Time-Normalized-Fitness-Swap E(15) = Random-Normalized-Fitness-Swap E(16) = Random-Time-Non Uniform Swap

Because the scale of the objective functions in each problem is different, they could not be compared directly. In order to determine the relative importance of each factor, two criteria are used in the literature: the *Signal-to-Noise ratio* (S/N) [13], and the *Relative Percentage Deviation* (RPD) [9, 14]. In the case of S/N ratio, the term 'signal' denotes the desirable value (response variable) and 'noise' denotes the undesirable value (standard deviation). Therefore, the aim is to maximize the S/N ratio. In the case of a minimization objective, the S/N ratio is calculated as in (20) [16].

$$S/N \text{ ratio} = -10 \log_{10} (\text{objective function})^2 \quad (20)$$

The second criterion, RPD , is used when the scale of the values of the objective functions are different from one problem to another and hence cannot be used directly in the analysis. the RPD is then calculated for each instance. The RPD is calculated as in (21).

$$RPD = \frac{Alg_{sol} - Min_{sol}}{Min_{sol}} \times 100 \quad (21)$$

Where Alg_{sol} and Min_{sol} are the obtained objective values for each replication of trial in a given combination and the obtained best solution, respectively. After converting the objective values to $RPDs$, the mean RPD is calculated for each trial. Six problems with different size are generated and used to discover

observed that the optimal solution of SC2SCN may contain more than $n+m-1$ loads and also, it may go up to $n+m$ loads. This conclusion complies with the conclusion arrived by Kowalski and Lev, (2008) [18] for the SFCTP.

3. Factorial Experiments

Due to the important affect of the factor levels in the performance of the AIS, in this section a calibration of the proposed AI2SCN algorithm through discovering the best level of each factor are presented. Also, In robust parameter design, the primary goal is to find factor settings that minimize response variation, while adjusting (or keeping) the process on target. After determining which factors affect variation, the settings for controllable factors are found that will either reduce the variation, make the AI2SCN algorithm insensitive to changes in uncontrollable (noise) factors, or both. Any algorithm designed with this goal will produce output that is more consistent. It can be observed that a number of factors affect the performance of the AI2SCN algorithm. The factors include the population size (A), the clone number (B), removable threshold ratio (C), the affinity function (D), and the mutation function (E) respectively. These factors and their levels are listed in Table 5. The levels considered for each factors are based on the empirical study using various combinations that observed to provide a good performance and the levels adapted in [16, 27] are also considered.

Table 5: Different Factors of the AI2SCN and Their Different Levels

Factor	Sym bol	Levels
Population size	A	A(1) = 35 A(2) = 40 A(3) = 45 A(4) = 50
Number of Clones	B	B(1) = 10 B(2) = 20 B(3) = 25 B(4) = 30
Remove Threshold (α)	C	C(1) = 5% C(2) = 10% C(3) = 20% C(4) = 30%
Affinity function	D	D(1) = $AF(P_k, P_r) = \sum_i y_i$

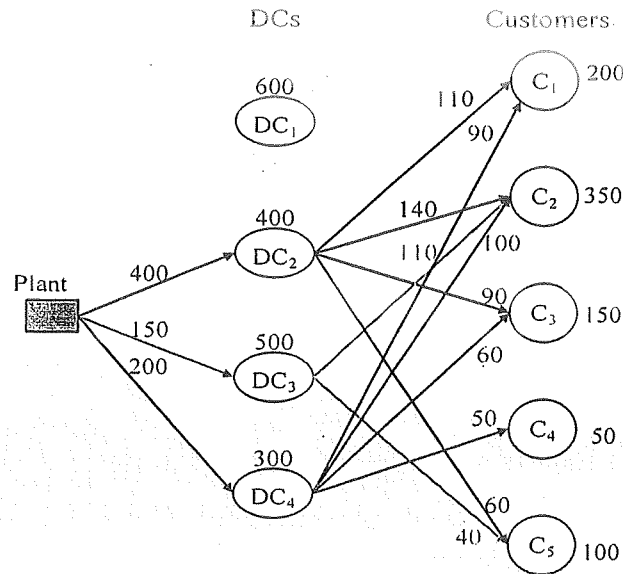


Fig. 6. Transported allocation for the illustrative example

Based on the local solution presented in Table 3 and Fig. 6 it can be observed that while DC1 is not selected to serve any customer, DCs 2, 3 and 4 are selected to serve all customers. The DC₂ serves customers 1, 2, 3 and 5, the DC₃ serves the customers 2 and 5, and the DC₄ serves customers 1, 2, 3 and 4. The total cost for this allocation, as presented in Table 3, is calculated as follows:

The shipping cost from the plant to DCs ($c_i \times x_i$) is $4 \times 0 + 2 \times 400 + 5 \times 150 + 3 \times 300 = 0 + 800 + 750 + 900 = 2450$

The opening cost of DCs is $f_{2,1} + f_{3,1} + f_{4,1} = 300 + 190 + 200 = 690$

The additional opening cost for DCs is $f_{2,2} + f_{3,2} + f_{4,2} = 400 + 0 + 200 = 600$

The opening cost of the route (i, j) – the opening cost from DCs to customers – is $f_{2,1,1} + f_{2,2,1} + f_{2,3,1} + f_{2,5,1} + f_{3,2,1} + f_{3,5,1} + f_{4,1,1} + f_{4,2,1} + f_{4,3,1} + f_{4,4,1} = 50 + 70 + 50 + 50 + 50 + 60 + 50 + 60 + 50 + 70 = 560$

The additional opening cost of the route (i, j) $f_{ij,2}$ is $= 0$

The shipping cost from DCs to customers is $c_{21} \times x_{21} + c_{22} \times x_{22} + c_{23} \times x_{23} + c_{25} \times x_{25} + c_{32} \times x_{32} + c_{35} \times x_{35} + c_{41} \times x_{41} + c_{42} \times x_{42} + c_{43} \times x_{43} + c_{44} \times x_{44} = 220 + 700 + 270 + 240 + 440 + 120 + 90 + 500 + 120 + 150 = 2850$

The total cost is $2450 + 690 + 600 + 560 + 0 + 2850 = 7150$.

Table 4: The cost items for the illustrative example

DC	$c_i \times x_i$	$f_{i,1}$	$f_{i,2}$	$\sum_{j=1}^m c_{ij} x_{ij}$	$f_{ij,1}$	$f_{ij,2}$
DC ₁	0	0	0	0	0	0
DC ₂	800	300	400	1430	220	0
DC ₃	750	190	0	560	110	0
DC ₄	900	200	200	860	230	0
	2450	690	600	2850	560	0

As illustrated in Table 3 and Fig. 6, while optimal solution of classical transportation problem has $n+m-1$ or less loads, the solution of the above illustrated example has ten loads that equal to $n + m + 2$. Therefore, it can be

	C_1	C_2	C_3	C_4	C_5	s_i	TS_i
DC ₁						600	600
DC ₂						400	400
DC ₃						500	500
DC ₄				50		300	250
d_i	200	350	150	50	100		1750
TD_i	200	350	150		100	800	

	C_1	C_2	C_3	C_4	C_5	s_i	TS_i
DC ₁						600	600
DC ₂						400	400
DC ₃						500	500
DC ₄	90			50		300	160
d_i	200	350	150	50	100		1660
TD_i	110	350	150		100	710	

	C_1	C_2	C_3	C_4	C_5	s_i	TS_i
DC ₁						600	600
DC ₂	110					400	290
DC ₃						500	500
DC ₄	90			50		300	160
d_i	200	350	150	50	100		1550
TD_i		350	150		100	600	

Continue as the same until reaching the total allocation as represented in the following table:

	C_1	C_2	C_3	C_4	C_5	s_i	TS_i
DC ₁						600	600
DC ₂	110	140	90		60	400	
DC ₃		110			40	500	350
DC ₄	90	100	60	50		300	
d_i	200	350	150	50	100		950
TD_i							

Output: The final allocation is:

C_j	4	1	1	2	2	2	3	3	5	5
DC_i	4	4	2	4	2	3	4	2	2	3
x_{ij}	50	90	110	100	140	110	60	90	60	40

Fig. 5. The result of applying the shipping procedure on the resulted S and D .

The capacities/supplies s_i and the demands d_j are illustrated by the network graph presented in Fig. 6. The local solution found by the proposed algorithm and the corresponding cost items are given in Table 3 and Table 4 respectively.

The population restored back to its original size through replacing the similar solutions with the new solutions that created randomly.

4. Illustrative example

In order to explain the characteristics of the proposed SC2SCN problem, an illustrative example with one plant, four DCs, and five customers is considered. The plant has no capacity limitation in production and each potential DC has distinct capacity in order to support the customers. The opening cost for using the route from the plant to each DC i is considered as $f_{i,1}$ and the additional cost is $f_{i,2}$ if the transported quantity through the same route exceeds the step load A_i . The opening cost for using the route from the DC $_i$ to the customer j is considered as $f_{ij,1}$ and the additional opening cost $f_{ij,2}$ if the transported units x_{ij} exceeds the step load A_{ij} for such route. These costs, the capacities and the step loads are given in Table 2. The objective is to minimize the total cost by selecting the DCs which fulfill demands of all the customers and the corresponding transported units x_{ij} . Also, in order to illustrate the solution of such problem, the main steps of the proposed algorithm are demonstrated as follows:

Table 2: The problem data of example 4×5

						1 2 3 4 5					1 2 3 4 5					1 2 3 4 5									
						d_i																			
						200 350 150 50 100																			
$f_{i,1}$	$f_{i,2}$	c_i	A_i	s_i	Variable cost c_{ij}					Fixed cost $f_{ij,1}$					Fixed cost $f_{ij,2}$					Step load A_{ij}					
500	600	4	300	600	5	3	2	4	6	90	60	80	60	70	100	150	180	140	160	100	80	120	70	110	
300	400	2	150	400	2	5	3	5	4	50	70	50	30	50	120	110	200	180	220	140	140	120	150	110	
190	500	5	250	500	3	4	6	5	3	80	50	70	90	60	200	100	100	130	150	120	120	130	140	120	
200	200	3	200	300	1	5	2	3	4	50	60	50	70	80	150	100	220	80	100	90	100	120	90	100	

First, the generated antibodies are decoded by applying the splitting procedure. The input of such procedure is the generated antibody and its outputs are the order of DCs S and the order of customers D . Fig. 4 exhibits the orders S and D resulted from applying the decoding procedure on the proposed antibody illustrated in Fig. 3.

D

4	1	2	3	5
---	---	---	---	---

 The customers' order

S

4	2	3	1
---	---	---	---

 The DCs' order

Fig. 4. The result of applying the decoding procedure

Second, the shipping procedure is applied on the result of splitting procedure in order to allocate the transported units x_{ij} that represent the feasible solution for SC2SCN problem. Thus the inputs of such procedure are the order of DCs S and the order of customers D and the output are the values of $x_{ij} \forall i$ and j . The result of applying the shipping procedure on the resulted S and D is presented Fig. 5.

solutions with high evaluation from being copied to the next generation and hence dominating the search [29]. This is technically applied to reduce the chance of a premature convergence to local optima. Many affinity measurements are used in the literature such as [30, 31]. Four different affinity functions are tailored to the AI2SCN algorithm to check the similarity between every two antibodies in a population. These affinity functions are illustrated in the following.

The first affinity function [27, 28] is based on counting the number of similar digits in the two solutions. The basic idea is that the more the number of similar digits in the two solutions is, the higher the similarity between them is. Based on a specific parameter, predefined parameter represents the number of similarities; the algorithm eliminates those solutions that have more similar digits than this parameter. This affinity function of two antibodies A_j and A_k is represented as in (16).

$$AF(P_k, P_r) = \sum_i y_i \quad (16)$$

$$\text{where } y_i = \begin{cases} 1 & \text{if the } i^{th} \text{ gene of } P_k = \text{the } i^{th} \text{ gene of } P_r \\ 0 & \text{Otherwise} \end{cases}$$

The other two techniques are based on the fitness of the two solutions. Although it is absolutely true for the SC2SCN problem that far different solutions might have the same fitness, these two techniques are applied and tested against the number of similarities technique for their simple calculations. The two functions evaluate the difference between the fitness of the two solutions. However, *the second affinity function* divides the difference by the minimum fitness of the two solutions as represented by (17) while the *third affinity function* divides the difference by the maximum fitness of the two solutions as represented by (18) and the fourth *affinity function* is calculated based on the 1 divides its fitness as represented by (19). Similar to the technique based on the number of similarities, the elimination of similar solutions is applied if the value calculated by any of these two functions is less than a specific threshold.

$$AF2 = \frac{MaxFitness - MinFitness}{MinFitness} \quad (17)$$

$$AF3 = \frac{MaxFitness - MinFitness}{MaxFitness} \quad (18)$$

$$AF4 = \frac{1}{Fitness} \quad (19)$$

		$NF = (LowestFitness - Fitness) / (LowestFitness - HighestFitness) \quad (9)$ $NS = MNS^{(1-(1-NF)^u)} \quad (10)$
13	Non-uniformity Swap	This function is designed to be inversely related with the ratio (T) of the current iteration number (CIN) and the total number of iterations (TNI). The NS for this MF is calculated by (11) where u is the degree of non-uniformity. $NS = MNS^{(1-T^u)}, \text{ where } T = CIN/TNI \quad (11)$
14	Time Normalized Fitness swap	This MF is based on both the time and the normalized fitness of the solution. The NS for this MF is calculated by (12) where T is the average of time and normalized fitness is calculated by (13). $NS = MNS^{(1-(1-TF)^u)} \quad (12)$ $TF = \frac{1}{2}(NF + (1 - T)). \quad (13)$
15	Random Normalized Fitness swap	A random factor $R \in [0, 1]$ is included so that the number of swaps is based on the non-uniform factor fitness but with some randomization. The NS for this MF is adopted as in (14) [13, 25]. $NS = MNS \times R^{(1-NF)^u} \quad (14)$
16	Random - Time -Non Uniform Swap	A random factor $R \in [0, 1]$ is included so that the number of swaps is based on the non-uniform factor and time fitness but with some randomization. The NS for this mutation is adopted as in (15) [13, 25]. $NS = MNS \times R^{(T^u)} \quad (15)$

3.4. Selection and Affinity Functions

The selection of the antibodies from one generation to the next one depends on some measurement of the affinity (similarity) among all the antibodies of the current generation [21, 22]. Affinity is the degree of binding between an antibody receptor and antigen. As affinity increases, binding increases, and thus, the immune recognition and response increase. [27]. Therefore, when solving SC2SCN problem using an immune algorithm, antibodies and antigen can be regarded as trial solutions and an optimal solution, respectively. The immune algorithm achieves an *optimal* solution by iteratively searching for the antibody with the highest affinity. For a detailed review of immunology and the artificial immune system see De Castro and Timmis [23, 24]. The calculations of the affinity between each two antibodies are applied to prevent similar

Table 1: Mutation Functions

# MF	Mutation Function	Description of MF
1	2PointSwap	Generate two random numbers j and k where $j, k \in [1, n + m]$, i.e. j and $k = \text{Rand}(1, n + m)$ and swap its corresponding genome [15, 27].
2	Inverse Swap	Generate two random numbers j, k where $j = \text{Rand}(1, n)$ and $k = \text{Rand}(1, n + m)$ and inverse the order of the antibody's digits between j and k [14, 18].
3	Neighbor swap	Generate a random number $j \in [1, n + m]$ and swapping the positions j and $j+1$. i.e. Generate $j = \text{Rand}(1, n + m)$ and swap the positions j and $j+1$.
4	Uniform-2Point Swap	Perform the 2PointSwap for a pre-defined number of times for each antibody during all iterations.
5	Random-2Point Swap	Generate a random number $NS \in [1, n + m]$ and performing the 2PointSwap NS times [15, 27].
6	Range Uniform Swap	Perform a uniform random number located in the range of 10% to 30% of the sum of problem dimensions $(m + n)$. The number of swaps $NS = \text{Rand}(1, r(m + n))$ where $r \in [0.1, 0.3]$.
7	Time-2Point Swap	Perform random-2PointSwap till a pre-defined ratio of time is elapsed followed by the 2PointSwap MF for the remaining time
8	Rand-Inverse Swap	Generate a random number $r \in [0, 1]$. If $r >$ pre-defined value v , then the <i>non uniform swap time</i> will be applied; else <i>Inverse Swap</i> MF will be applied.
9	Time-Inverse Swap	A random number of swaps is applied till the time passes a pre-defined ratio and apply the Inverse Swap MF for the remaining time.
10	Rand-Neighbor Swap	Generate a random numbers $r \in [1, (n + m)/2]$ and repeating <i>Neighbor Swap</i> r times for each antibody A_p .
11	Time-Neighbor Swap	A random number of Neighbor Swap MF is applied till a pre-defined ratio of time elapsed and the Non Uniform Swap MF is applied for the remaining time [19].
12	Normalized Fitness swap	This MF is designed to be directly related with the Normalized Fitness (NF) of the solution as the SC2SCN is a minimization problem [25]. The normalized fitness of each antibody is calculated using (9) and the NS for this MF is adopted by (10).

2.4. If $d_j = s_i$ and $s_i \leq A_{ij}$, set $x_{ij} = s_i$, $s_i = 0$, $d_j = 0$, $TD = TD - s_i$, remove $S(L)$, $L = 0$, $TDT = TD$, and remove $D(1)$,

2.5. If $d_j = s_i$ and $s_i > A_{ij}$, set $j = D(1)$, set $i = S(L)$, $x_{ij} = A_{ij}$, $s_i = s_i - A_{ij}$, $d_j = d_j - A_{ij}$, $TD = TD - A_{ij}$, and $TDT = TD - s_i$.

2.6. If $d_j > s_i$ and $s_i > A_{ij}$ and $(TDT - s_i) \geq (d_j - A_{ij})$, set $s_i = s_i - A_{ij}$, $d_j = d_j - A_{ij}$, $x_{ij} = A_{ij}$, $TD = TD - A_{ij}$, and $TDT = TD - s_i$.

2.7. If $d_j > s_i$ and $s_i \leq A_{ij}$, set $d_j = d_j - s_i$, $x_{ij} = s_i$, $TD = TD - s_i$, $TDT = TD - s_i$, remove $S(L)$, and $L = L - 1$.

2.8. Update $L = L + 1$

Step 3. Repeat steps 2.1 to 2.8 until $|S| < L$ or $|D| < 1$.

Step 4. If $L = |S|$ or $|D| = 1$, set $j = D(1)$ and $i = S(L)$. One of the following states will occur:

II. If $d_j < s_i$, set $s_i = s_i - d_j$, $x_{ij} = d_j$, $TD = TD - d_j$, remove $D(1)$, and $L = 0$.

III. If $d_j > s_i$, set $d_j = d_j - s_i$, $x_{ij} = s_i$, $TD = TD - s_i$, remove $S(L)$, and $L = L + 1$.

IV. If $d_j = s_i$, set $x_{ij} = s_i$, $s_i = 0$, $d_j = 0$, $TD = TD - s_i$, remove $S(L)$, remove $D(1)$, and $L = 0$ }

Step 5. Update $TDT = TD$

Step 6. Repeat steps 2 to 5 until $|S| + |D| \leq 1$.

Step 7. Return $x_{ij} \forall i = 1, 2, \dots, m$ and $j = 1, 2, \dots, n$.

Where $|Q|$ is the function that returns the length of queue Q .

The third phase is calculating the fitness of each antibody through evaluating the solution resulted from the shipping procedure by using the objective function Z (6). The resulted objective function value is considered as the total cost of shipping the units to the customers. This value is assigned to the corresponding antibody as its fitness.

3.3. Cloning and mutation

In the generated population, each antibody is cloned CN number of times, where CN denotes the Cloning Number. The clones are then mutated to rearrange the structure for generating new antibodies that are different from their parents. In the proposed AI2SCN algorithm, sixteen Mutation Functions (MFs) are used [16]. The first three basic Mutation Functions (MFs) are namely: two point swap (2PointSwap) [13, 25], inverse swap (InvSwap) [12, 15], and neighbor swap (NeibSwap) [36]. In addition, the other thirteen are hybrid MFs. These MFs are illustrated in Table 1.

The details of the main steps are elaborated in the following subsections.

3.1. Antibody Structure and initialization

The solution (antibody) representation is an important issue in designing the AIA. In the proposed AI2SCN algorithm, the length of each antibody P_k is considered equal to the sum of the problem dimensions, viz., the DC numbers represented by the integer numbers from 1 to m and the customers by integers from $m+1$ to $m+n$. Hence, the proposed coding scheme (antibody structure) consists of the set of random unrepeated integers in the interval $[1, m+n]$ in any sequence; where the length of the scheme is equal to $m+n$. Fig. 3 depicts a sample antibody which is used to code a 4x5 FCTP. As shown in Fig. 3, the cells contain unrepeated integers between 1 and 4+5. The population is initialized randomly by creating l antibodies P_k ($k=1$ to l) where l represents the population size.

8	3	5	6	4	7	2	1	9
---	---	---	---	---	---	---	---	---

Fig. 3. An example of proposed antibody structure

3.2. Evaluation of the Solutions

Evaluate each of each antibody as illustrated in Fig. 2 can be done through three phases as follows:

The first phase is applying the decoding procedure [20]. This procedure is used to split each generated antibody P_k into two antibodies, one representing the order of DCs referred by S and the other representing the customers' order D . The inputs of this procedure are the generated antibody P_k , the number of suppliers m , and the number of customers n while the results are the sequences of DCs S and customers' D .

The second phase is applying the Shipping procedure. This procedure finds a feasible solution for SC2SCN problem based on the outputs of the splitting procedure. Thus the inputs are the order of DCs S and the order of customers D (the output of splitting procedure). Based on these orders, the shipping procedure allocates units x_{ij} (feasible solution) of the SC2SCN. This procedure guarantees the validity of both the first and the second constraints, denoted by (7) and (8) respectively. The Pseudo code for the shipping procedure is as follows:

Step 1. Set $TD = \sum_{j=1}^n d_j$ and $TDT = TD$. i.e. TD and TDT are set to equal the total demand.

Step 2. Set $L = 1$.

2.1. Set $j = D(1)$ and $i = S(L)$.

2.2. If $d_j < s_i$ and $d_j \leq A_{ij}$, set $s_i = s_i - d_j$, $x_{ij} = d_j$, $TD = TD - d_j$, $TDT = TD - s_i$, remove $D(1)$, and $L = 0$,

2.3. If $d_j < s_i$ and $d_j > A_{ij}$, set $s_i = s_i - A_{ij}$, $d_j = d_j - A_{ij}$, $x_{ij} = A_{ij}$, $TD = TD - A_{ij}$, and $TDT = TD - s_i$.

adding two main procedures; the first being the procedure for splitting the antibody into two orders, one of them to represent the customers' order and the other to represent the suppliers' order and the second being the allocating procedure that is used for finding the corresponding feasible solution of these orders. The proposed AI2SCN algorithm preserves the essential principles of *Opt-aiNet* [23, 24] algorithm including the cloning, mutation, and clone selection. The proposed AI2SCN algorithm for solving SC2SCN is adjusted and adopted the Altassan [20] and El-Sherbiny algorithms [19] by replacing the allocation procedure with the proposed shipping procedure which will be used for defining the units x_{ij} shipped through each route (i, j) and by backward calculates the quantity x_i shipped from the plant to each DC_i . The main steps of the proposed AI2SCN algorithm for solving SFCTPs are presented in Fig. 2.

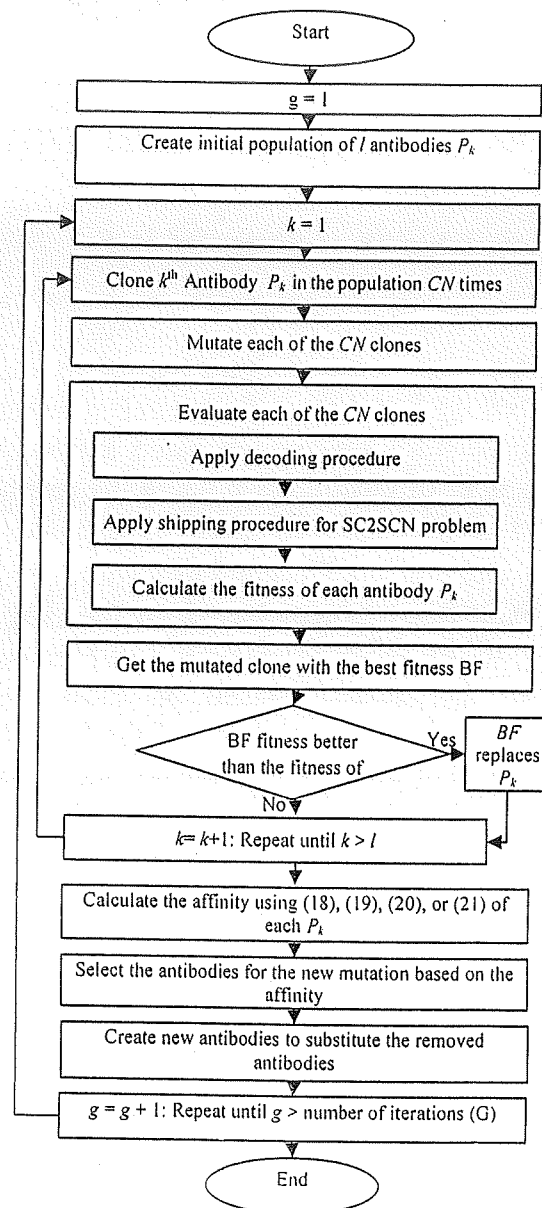


Fig. 2. Main architecture of the AI2SCN Algorithm

$$y_{i,1} = \begin{cases} 1 & \text{if } \sum_{j=1}^n x_{ij} > 0 \quad \forall i \\ 0 & \text{otherwise} \end{cases}$$

$$y_{i,2} = \begin{cases} 1 & \text{if } \sum_{j=1}^n x_{ij} > A_i \quad \forall i \\ 0 & \text{otherwise} \end{cases}$$

Therefore, the stander mathematical model of the proposed SC2SCN problem can be constructed by substituting (4) and (5) in the CFCTP mathematical model as follows:

$$\text{Min } Z = \sum_{i=1}^m \sum_{j=1}^n (c_{ij} x_{ij} + y_{i,1} f_{ij,1} + y_{i,2} f_{ij,2}) + \sum_{i=1}^m \left(c_i \sum_{j=1}^n x_{ij} \right) + \sum_{i=1}^m (y_{i,1} f_{i,1} + y_{i,2} f_{i,2}) \quad (6)$$

$$\text{s.t. } \sum_{i=1}^m x_{ij} = d_j \quad \text{for } j = 1, \dots, n \quad (7)$$

$$\sum_{j=1}^n x_{ij} \leq s_i \quad \text{for } i = 1, \dots, m \quad (8)$$

$$x_{ij} \geq 0 \quad \forall i, j$$

$$y_{i,1} = \begin{cases} 1 & \text{if } x_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j$$

$$y_{i,2} = \begin{cases} 1 & \text{if } x_{ij} > A_i \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j$$

$$y_{i,1} = \begin{cases} 1 & \text{if } \sum_{j=1}^n x_{ij} > 0 \quad \forall i \\ 0 & \text{otherwise} \end{cases}$$

$$y_{i,2} = \begin{cases} 1 & \text{if } \sum_{j=1}^n x_{ij} > A_i \quad \forall i \\ 0 & \text{otherwise} \end{cases}$$

The following relationships can be observed among the Step FCTP (SFCTP), the Capacitated FCTP (CFCTP), and Step Capacitated FCTP (SCFCTP) - SC2SCN:

If the $\sum_{j=1}^m A_j \geq s_i, \forall i$ and $\sum_{i=1}^n A_i \geq d_j, \forall j$, the optimal solution of SC2SCN is an optimal solution of SC2SCN.

If the plant capacity is greater than or equal to the $\text{Min}(\sum_{i=1}^n s_i, \sum_{j=1}^m d_j)$, the optimal solution of SC2SCN is an optimal solution of SFCTP.

If the plant capacity is greater than or equal to the $\text{Min}(\sum_{i=1}^n s_i, \sum_{j=1}^m d_j)$, the optimal solution of SFCTP is an optimal solution of SC2SCN.

3. The Proposed AI2SCN Algorithm

In this section, an algorithm has been developed for solving SC2SCN problem using artificial immune algorithm (AIA). Altassan et al. [20] and El-Sherbiny [19] applied AIA for solving FCTPs and SFCTPs respectively by

2. The SC2SCN description and formulation

CFCTP can be described as a distribution problem in which there are m suppliers (warehouses or factories) and n customers (destinations or demand points). Each of the m suppliers can ship to any of the n customers at a shipping cost per unit c_{ij} (unit cost for shipping from supplier i to customer j) plus a fixed cost f_{ij} , assumed for opening this route. Each supplier $i = 1, 2, \dots, m$ has s_i units of supply and each customer $j = 1, 2, \dots, n$ demands d_j units. x_{ij} is the quantity to be transported from plant i to customer j through the route (i, j) . The objective is to determine which routes are to be opened and the size of the shipment, so that the total cost of meeting demand, given the supply constraints, is minimized. The stander mathematical model of CFCTP can be represented as follows [16]:

$$\text{Min } Z = \sum_{i=1}^m \sum_{j=1}^n (c_{ij}x_{ij} + f_{ij}y_{ij}) + \sum_{i=1}^m (c_i \sum_{j=1}^n x_{ij}) + \sum_{i=1}^m f_i y_i \quad (1)$$

$$\text{s.t. } \sum_{i=1}^m x_{ij} = d_j \quad \text{for } j = 1, \dots, n \quad (2)$$

$$\sum_{j=1}^n x_{ij} \leq s_i \quad \text{for } i = 1, \dots, m \quad (3)$$

$$x_{ij} \geq 0 \quad \forall i, j$$

$$y_{ij} = \begin{cases} 1 & \text{if } x_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad \forall i, j$$

$$y_i = \begin{cases} 1 & \text{if } \sum_{j=1}^n x_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad \forall i$$

In SFCTP the fixed cost f_{ij} , for route (i, j) is proportional to the transported amount through this route [18]. So, an additional cost is added when the transported units exceeds a certain amount A_{ij} . Assume that $f_{ij,1}$ is the fixed cost to open a route (i, j) if the shipped units is less than or equal to A_{ij} and when it exceeds A_{ij} an additional fixed cost to open the same route is $f_{ij,2}$. These costs can be calculated by (4).

$$f_{ij} = y_{ij,1} f_{ij,1} + y_{ij,2} f_{ij,2} \quad (4)$$

where

$$y_{ij,1} = \begin{cases} 1 & \text{if } x_{ij} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ij,2} = \begin{cases} 1 & \text{if } x_{ij} > A_{ij} \\ 0 & \text{otherwise} \end{cases}$$

Also, assume that $f_{i,1}$ is the fixed cost to open a DC_i if the shipped units is less than or equal to A_i and when it exceeds A_i an additional fixed cost to open the same DC is $f_{i,2}$ is add. These costs can be calculated by (5).

$$f_i = y_{i,1} f_{i,1} + y_{i,2} f_{i,2} \quad (5)$$

where

production and each potential DC has distinct (different) capacity in order to support the customers. A shipping cost from the plant to DCs and an opening cost for each potential DC are considered. Also, the opening cost is considered to be proportional to the amount shipped from each DC to each customer. These opening costs' structure cause the value of the objective function to behave like a step function. (i.e. An additional cost is added when the transported units exceeds a certain amount). This rule applied on both shipped units; the shipped units from the plant to each DC and the shipped units from the DCs to each customer. The proposed model minimizes the total cost of shipping by selecting the DCs through which the products are supplied in order to fulfill the demands of all the customers. The total cost considers shipping cost from plant to DCs, opening cost for each DC, shipping cost from DCs to the customers, and the opening cost for shipping from DCs to customers where the opening cost is depending on the amount shipped.

Since problems with fixed charge are usually NP-hard (Nondeterministic Polynomial-time) [6], the computational time to obtain exact solutions increases in a polynomial fashion and very quickly becomes extremely difficult and long as the size of the problem increases. In the case of the SC2SCN due to the step function structure of the objective function Z , we are dealing with a "NP-hard" problem with much "higher degree" of the polynomial complexity.

This paper aims to introduce the description, the mathematical formulation and the solution algorithm for the proposed SC2SCN problem. An artificial immune algorithm for handling such NP-hard problem (AI2SCN) has been proposed to provide an optimal or near optimal solution. In order to calibrate the proposed AI2SCN algorithm, six problems with different sizes have been solved to select its best setup.

The rest of the paper is organized as follows: in section 2, SC2SCN problem description and its mathematical model are presented. In section 3, an illustrative example of the proposed SC2SCN problem is presented. In section 4, the proposed AI2SCN algorithm is described, and parametric analysis for the AI2SCN algorithm is carried out in section 5. Finally, the conclusion and future work are reported in section 6.

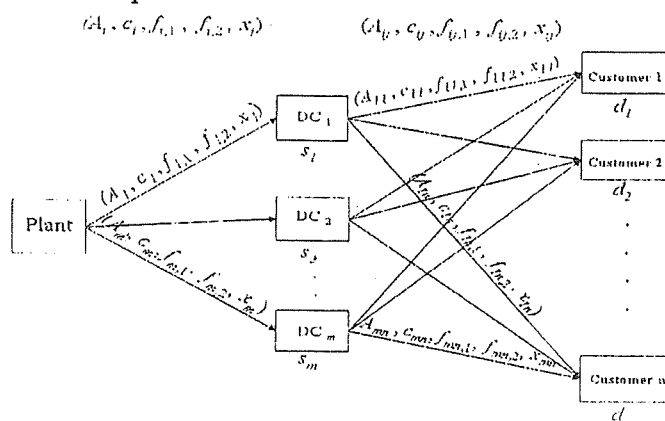


Fig. 1. Step capacitated two-stage supply chain network (SC2SCN)

Notations and symbols

A_i	The step load i ; the limit of the transported units from the plant to DC_i to apply the additional opening (fixed) cost $f_{i,2}$ for using the route from the plant to DC_i
A_{ij}	The step load ij ; the limit of the transported units through the route (i, j) to apply the additional opening (fixed) cost $f_{ij,2}$ for using the route (i, j) from DC_i to customer j
c_i	Shipping (variable) cost from the plant to DC_i
c_{ij}	Shipping (variable) cost per unit from DC_i to customer j
C_j	Customer j
DC_i	Distribution Center i
d_j	Demand of customer j
f_i	Fixed cost for opening DC_i if $x_i > 0$; it will take values $f_{i,1}$ and $f_{i,2}$ depending on the value of x_i
$f_{i,1}$	Opening (fixed) cost for using DC_i if $x_i > 0$
$f_{i,2}$	Additional opening (fixed) cost for using DC_i if $x_i > A_i$
f_{ij}	Fixed cost for opening route (i, j) if $x_{ij} > 0$; it will take values $f_{ij,1}$ and $f_{ij,2}$ depending on the value of x_{ij}
$f_{ij,1}$	Opening (fixed) cost of using the route (i, j) if $x_{ij} > 0$
$f_{ij,2}$	Additional opening (fixed) cost of using route (i, j) if $x_{ij} > A_{ij}$
g	Generation number identifier ($g=1$ to G)
G	The number of generations
i	Distribution center identifier ($i=1$ to m)
j	Customer identifier ($j=1$ to n)
k	Antibody identifier ($k=1$ to l)
l	Antibody population size
m	Number of distribution centers
n	Number of customers
P_k	The antibody k
RPD	Relative percentage deviation
S/N	Signal noise ratio
s_i	Capacity of DC_i
TD	The total demand
x_i	The quantity to be transported from the plant to DC_i
x_{ij}	The quantity to be transported from distributor i to customer j through the route (i, j)
Z	The total cost of distribution

In this paper, a new version of the FCTP is proposed in the form of Step Capacitated FCTP. This problem will be denoted as Step Capacitated Two-stage Supply Chain Network (SC2SCN) and is presented in Fig. 1. In this problem, the situations considered represent presence of potential distribution centers (DCs) prior to distribution of product units to the customers from the plant. Each of the potential DC is served from the plant and can ship items to any of the customers. It is assumed that the plant has no capacity limitation in

In such problem, one of the strategic decisions is the allocation of transported quantities from production plant to distribution centers (DCs) and from the DCs to the customers in a cost effective way.

Usually, unit cost of transportation is used to find the optimal distribution schedule. A basic assumption in any transportation problem is that the cost of transportation is directly dependent on the number of units transported [3]. Many practical transportation and distribution problems can be modeled as fixed cost transportation problems FCTPs [4, 5, 6]. Considerable work has been done in solving FCTP, such as, Tabu search method [7], genetic algorithms (GA) based on a spanning tree with Prüfer numbers [8, 9, 10], and genetic algorithms based on a matrix permutation representation [11] and [12] have improved the effective coding of the spanning tree method based on edge sets. The genetic algorithm creates a sorted set of edges to encode the spanning tree, which is efficient compared with evolution strategies at a certain level [13]. Moreover, to improve solution quality, Hajiaghahi et al. [9, 10] addressed a nonlinear FCTP using a spanning tree based genetic algorithm. Nevertheless, the quality of solutions attained largely depends on the randomness. Also, a hybrid particle swarm method with artificial immune learning used for solving FCTP [14], a novel artificial immune algorithm for solving FCTP [15].

Many versions of FCTPs have been proposed; such as, Capacitated FCTP (CFCTP) [16, 17] and Step FCTP (SFCTP) [18]. In CFCTP each potential distribution center DC has distinct capacity in order to support the customers and also transportation cost from the plant to DCs and an opening cost for each potential DC are considered. CFCTP has been solved by using artificial immune and genetic algorithms [16, 17]. The other version is the SFCTP where the fixed cost is incurred for every route that is used in the solution and it is dependent on the amount shipped. This cost structure causes the value of the objective function to behave like a step function.

SFCTP is considered to be “NP-super hard” problem with much “higher degree” of the polynomial complexity [18]. However, not much research has been carried out in the area of SFCTP. The SFCTP been solved by using artificial immune [19] and heuristic method [20].

On the other hand, some special Artificial Immune Systems (AIS) are developed to solve complex optimization problems. One of them is aiNet [21, 22] that is inspired by biological immune system. Opt-aiNet [21] is an application of aiNet in function optimization. Opt-aiNet [23, 24] considers the optimized objective function as antigen, and the candidate solutions as antibodies. The candidate antibodies evolve according to the matching degree between antibodies and antigen that is fitness. The better the matching between the antibodies is the less the mutation degree of candidate antibody, and vice versa. AIS has been used in many applications such as job shop scheduling problems [25, 26].

*Step Capacitated Two-Stage Supply Chain Network:
Application of Artificial Immune Algorithm*

Mahmoud M. El-Sherbiny¹

Khalid M. Altassan²

Abstract

In this paper a new version of the fixed charge transportation problem (FCTP) is proposed in the form of Step Capacitated Two-Stage Supply Chain Network in which a given number of intermediate distribution centers (DCs) are located between the supply plant and the customer locations. The DCs are considered to be capacitated in order to fulfill the demand of each customer. In addition, the opening cost for each potential DC and each route are considered. While the previous studies for capacitated FCTP considered that opening cost for each DC and each route to be fixed, the problem considered is the one where the opening costs depend on the amount shipped. This cost dependency structure causes the value of the objective function to behave like a step function. This problem will be denoted as Step Capacitated Two-stage Supply Chain Network (SC2SCN). The description and the mathematical model of such super NP-hard problem are presented. In order to solve such problem, this paper proposes an artificial immune algorithm referred to as AI2SCN. The proposed AI2SCN algorithm, guarantees the feasibility of all the generated antibodies. Due to the significant role of AI2SCN's factors on the performance quality, relative percentage deviation (*RPD*) and signal noise ratio (*S/N ratio*) methods are applied in order to emphasize the factors with the highest impact, calibrate and fine tune the best setup for the AI2SCN algorithm.

Keywords: *Step capacitated two-stage supply chain network; Fixed-charge transportation; Convergence; Artificial immune; Mutation function; Affinity function.*

1. Introduction

The cost of a product distribution plays a crucial role in the determination of its price and it accounts for about 30% of its cost. Hence, the problem of product distribution (transportation) is an important concern for industrial firms that have supply chain networks [1]. The transportation network design is one of the most important fields of supply chain management that offers a great potential to reduce costs. The transportation problem (TP) is a well-known basic network problem that was originally proposed by Hitchcock [2].

¹ Operations Research Dept., Institute of Statistical Studies & Research (ISSR), Cairo University, Egypt.
msherbiny@cu.edu.eg

² King Saud University, Faculty of Business Administration, Saudi Arabia, kaltassan@ksu.edu.sa

Index

Operations Research

1	Step Capacitated Two-Stage Supply Chain Network: Application of Artificial Immune Algorithm Mahmoud M.El-Sherbiny , Khalid M. Altassan	1-25
2	On the Achievement Stability Set for Parametric Linear Goal Programming Problem Osman. M.S , Maaty.M.A and Farahat.F.A	26-35
3	On Duality of Multiobjective Rough Convex Programming Problems M.S.A. Osman , M.H.Eid and M.A.Elsisy	36-46
4	An Interactive Approach for Solving Multi- Objective Nonlinear Programming Problems with Fuzzy Parameters E.E.Ammar , M.L.Hussein and H.A.Khalifa	47-59
5	Study on Possibilistic Multiobjective Solid Transportation Problems E .E . Ammar , M .L .Hussein and H .A .Khalifa	60-76
6	A Constraint Programming Approach to Optimize Coal Mining Rail Operations Mahmoud Masoud	77-93
7	On Multi-Step Fixed Charge Transportation Problem Ibrahim A. Aljasser, Mahmoud M .El-Sherbiny and Bokkasam Sasidhar	94-106
8	Simulation of Dynamic Project Management Optimization M .M .Ghada , A .M .Atef , G .H .Mohamed and E .H .El Hamy	107-117
9	Solving the Transportation problems Using the Hungarian Algorithm Abo Shady. A. Mona , Ragab. M.Aly and El-Sherbiny.M. Mahmoud	118-128
10	A Modified Version of the Improved Zero Point Method for Solving Transportation problems Abo Shady. A. Mona , Ragab. M .Aly and El-Sherbiny.M. Mahmoud	129-147
11	Treating Multi-Objective problems Using Special Approximations S .H .Eldeeb , M. H. Gadallah , O .S .Soliman and H.A.Khalifa	148-163



Cairo University



Institute of Statistical Studies and Research

**The 48th Annual Conference on Statistics , Computer
Sciences and Operation Research**

Operations Research

23-26 Dec ,2013