International Journal of
Mathematics And its Applications
ISSN: 2347-1557

Int. J. Math. And Appl., **11(1)**(2023), 97–104
Available Online: http://ijmaa.in

A New Hybrid Ant-Based Approach to the Economi Triangulation Problem

Shaleen Begum[1,*], R. S. Patel[1]

[1]*Department of Mathematics, Government Autonomous P.G. College, Satna, Madhya Pradesh, India*

Abstract

In the study of economical problems and to find their solutions, Triangulation plays an important role. Input-Output Matrices has been intensively studied in order to understand the complex series of interactions among the sectors of an economy. The problem refers to finding a simultaneously permutation of rows and columns of a matrix such as the sum of the entries which are above the main diagonal is maximum. This is a linear ordering problem – a well-known NP-hard combinatorial optimization problem. In this paper a new ant algorithms is proposed to efficiently solve the triangulation problem. Starting from a greedy solution, the proposed model hybridizes the New Ant Colony System (NACS) metaheuristic with an Insert-Move (IM) local search mechanism able to refine ant solutions. We also tested NACS-IM algorithm some real-life economic data sets.

Keywords: Linear Ordering Problem; Heuristics; New Ant Colony Optimization.

1. Introduction

To find the optimum solution in various economical problems is highly investigated domain. Some of these problems are so complex and they cannot be solved by simple technique. For this types of problems we use triangulation technique. The development of metaheuristics able to efficiently find high-quality near-optimal solutions to such problems with reasonable computational effort is essential. A special class of combinatorial optimization problems refers to the linear ordering problems where the objective is to find a permutation with optimum cost of some objects. One such problem is the Triangulation Problem for Input-Output Matrices seeking for a permutation of rows and columns in a given matrix of weights W such that the sum of the entries above the main diagonal (of the permuted matrix) is maximized. This problem has been intensively studied by economists [13] in order to understand the interdependencies and complex series of transactions among the sectors of an economy [2]. A variety of models from exact methods to heuristics for solving this NP-hard problem can be found in the literature [16]. To the best of our knowledge, this paper presents the first attempt

*Corresponding author (dranimeshgupta10@gmail.com)

to heuristically approach LOP using new ant-based models. The proposed model integrates New Ant Colony Optimization (NACO) with local search heuristics to solve LOP. The initial solution is generated in a greedy manner and is used as the starting point for a New Ant Colony System (NACS). Furthermore, the ant-based model is hybridized with a local search mechanism based on insert-moves able to improve the solutions. The proposed hybrid new ant-based model is tested for addressing the instances available in the real world LOLIB [12] and SGB [10] libraries.

2. The Triangulation Problem for Input-Output Tables

The Triangulation Problem for Input-Output Matrices is a Linear Ordering Problem (LOP), a well-known NP-hard combinatorial optimization problem [16]. The economy of a country or region is divided in sectors and an input-output table is built with entries quantifying the transactions between any two sectors. The triangulation of the input-output matrix means the detection of an optimal hierarchy of economic sectors (suppliers should be predominantly arranged before consumers in the matrix). Using graph theory terms LOP can be formulated as the problem of searching for an acyclic tournament having the maximum sum of arc weights in a complete weighted graph [1]. Let $G = (\mathcal{V}, \mathcal{A}, w)$ be a weighted graph where $\mathcal{V}$ represents the set of vertices, A contains the arcs of the graph and the function w refers to the weights associated with arcs. LOP aims to maximize the following functional:

$$C_G(\pi) = \Sigma_{i \neq j, i \preceq j} w((\pi(i), \pi(j)) \quad (1)$$

where π is a permutation of $\mathcal{V}, i, j \in \mathcal{V}$ and $\preceq$ is a total order relation on $\mathcal{V}$. Existing approaches to solve LOP include integer programming methods like branchand-cut [16], branch-and-bound [9], and interior point method [14]), approximate algorithms [8,15]. Heuristics such as tabu search [6], scatter search [7], iterated local search [11], variable neighborhood search [5] and evolutionary strategies [17] have been successfully employed.

3. The Proposed Hybrid New Ant-based Model

We propose a hybrid model based on the New Ant Colony System (NACS) metaheuristic [4] for solving the Triangulation Problem. Starting from a greedy initial solution, NACS is further hybridized with a particular local search mechanism for solution refinement. Ant algorithms are designed to solve optimization and distributed control problems by replicating the behavior of social insects to the search space. In the ACS model [4], each ant generates a complete tour (associated to a problem solution) by probabilistically choosing the next vertex based on the cost and the pheromone intensity on the connecting arc.

The problem is represented as a complete directed graph with n vertices; the function w assigns real values to arcs defining the static matrix of weights $W = (w_{ij}), 1 \preceq i, j \preceq n$. Besides the weight, each arc is associated with a pheromone intensity (built up by ants during the search). The pheromone matrix τ is dynamic and has the same dimensions as W.

A problem solution (a permutation of vertices) is a list of n vertices constructed by each artificial ant by moving from one vertex to another. Each ant keeps the list of already visited vertices in order to further avoid them (this is an element of tabular search integrated with the hybrid ant-based model).

3.1 Greedy solution initialization

One element of hybridization within the proposed ant-based method for LOP refers to the usage of a greedy approach to build the initial solution for each ant. For a complete solution π (meaning a permutation of length n), the neighborhood $N(\pi)$ contains the permutations that can be obtained from π by left compounding with a transposition. This means that a neighbor for a permutation is obtained by interchanging two of its elements. Let o denote the compounding operator for permutations and (ij) refer to a transposition. $N(\pi)$ has $\frac{n(n-1)}{2}$ elements and is given below:

$$N(\pi) = \{\sigma|(kr) : \sigma = (\pi(k)\pi(r)) \circ \pi \text{ and } 1 \preceq k < r \preceq n\}. \tag{2}$$

When moving from a permutation π to its neighbor σ the objective function has an added value of:

$$\begin{aligned} diff_{1 \preceq k < r \preceq n}(k, r) \quad = \quad & \max\{w_{\pi(i)\pi(j)}, w_{\pi(k)\pi(r)}\} \\ & + \sum_{i=k+1}^{r-1} [\max\{w_{\pi(r)\pi(i)}, w_{\pi(i)\pi(k)}\} - \min\{w_{\pi(i)\pi(r)}, w_{\pi(k)\pi(i)}\}] \end{aligned} \tag{3}$$

The greedy initial local search procedure chooses the best-improvement move – the one with the highest value for $diff(k, r)$ given by 3 – when investigating the neighborhood of a permutation. The proposed ant-based algorithm is initialized with the greedy solution obtained during this initial search. The pheromone matrix τ is initialized with $\tau_{ij} = \tau_0, 1 \preceq i, j \preceq n$, where τ_0 is a small positive constant. The number of artificial ants is denoted by m (a constant value).

3.2 Solution construction and pheromone update

Let us denote by i the current vertex for an ant k. The next vertex j is selected according to the pseudo-random proportional rule [3] given by:

$$j = \begin{cases} arg \quad \max_{l \in N}\{\frac{\tau_{il}}{w_{il}^{\beta}}\} & \text{if } q \preceq q_0 \\ J & \text{otherwise} \end{cases} \tag{4}$$

where, q is a random variable uniformly distributed in $[0,1]$, N_i^k refers to the feasible neighborhood of $ant \quad k$ from vertex i, q_0 and β are parameters, and J is a random variable having the following probability distribution [3,4]:

$$p_{ij}^k = \frac{\tau_{ij}/w_{ij}^{\beta}}{\sum\limits_{l \in N_i^k} \tau_{il}/w_{il}^{\beta}} \tag{5}$$

Based on the NACS model, pheromone update occurs in both the online and offline phases. For the online phase, the pheromone is updated during the solution construction, immediately after an ant crosses the $arc(ij)$, based on the following updating rule [3]:

$$\tau_{ij} = (1-\xi)\tau_{ij} + \xi\tau_0 \tag{6}$$

where, ξ and τ_0 are model parameters. After all ants have constructed a solution, the offline phase implies only the arcs from the best-so-far solution and uses the following formula [3]:

$$\tau_{ij} = (1-\rho)\tau_{ij} + \rho/C_{bs} \tag{7}$$

where, C_{bs} is the cost of the best-so-far solution, and ρ is a parameter.

3.3 Insertion-based local search

The proposed new ant-based model for LOP is hybridized with a local search mechanism based on insertions aiming to further improve and refine the solution. The local search mechanism is based on the neighborhood search proposed for LOP [5]. Insert moves (IM) are used to create a neighborhood of permutations for one solution. An insert move for a permutation π at $\pi(j)$ and i means deleting the element from position j and inserting it between elements $\pi(i-1)$ and $\pi(i)$. This operation results into a permutation σ (obtained from π) for which the objective function (for $i \preceq j$) is the following:

$$c_G(\sigma) = c_G(\pi) + \sum_{k=1}^{j-1} \max\{w_{\pi(j)\pi(k)}, w_{\pi(k)\pi(j)}\} \tag{8}$$

For a complete solution π a neighborhood $N(\pi)$ based on the IM mechanism is defined as follows:

$$N(\pi) = \{\sigma : \sigma = insert\_move(\pi(j), i); i = 1, 2, \ldots, j-1, j+1, \ldots n\} \tag{9}$$

3.4 Algorithm description

The proposed model for solving LOP starts with a greedy search. NACS-based rules are applied and a local search mechanism based on insert moves is engaged. The resulting algorithm is called New Ant Colony System-Insert Move (NACS-IM) and is outlined below.

NACS-IM Algorithm for solving LOP

procedure GreedyInitialSearch – using 2,3

procedure NACS-IM

Set parameters, initialize pheromone trails

while (termination condition not met)

  do ConstructAntsSolutions – using 4,5,6

Apply Insert-Moves – using 8,9

UpdatePheromones – using 7

end

4. Numerical Results

The proposed NACS-IM algorithm is engaged for the 45 problem instances from the real-world LOLIB library [12] and the 25 instances from the SGB library [10]. LOLIB contains input-output tables from Indian economy sectors while SGB includes input-output tables from the States economy. The parameters considered for ACS-IM are: $\beta = 3, \tau_0 = 0.01, \rho = 0.01, q_0 = 0.9$ and the number of ants is equal to the number of vertices. The results obtained for LOLIB are presented in Table 1 (NACS-IM with five runs of 50000 iterations). Table 1 shows the reported optimum solution for each instance, the best solution obtained by NACS-IM, the number of optimum solutions reported by NACS-IM as well as the average deviation of the obtained solution from the optimum one.

No.	Instances	Optimal	NACS-IM		
			Max.	No. Opt.	Avg, Dev.
1	60-stabu3	642050	640424	0	0.0027
2	60-stabu2	627926	627560	0	0.0027
3	60-stabu1	422088	421454	0	0.0026
4	56-tiw56r72	341623	341623	15	0.0011
5	56-tiw56r67	270497	270497	14	0.0039
6	56-tiw56r66	256326	256326	25	0.0009
7	56-tiw56r58	160776	160776	3	0.0016
8	56-tiw56r54	127390	127390	1	0.0019
9	56-tiw56n72	462991	462991	3	0.0017
10	56-tiw56n67	277962	277962	5	0.0045
11	56-tiw56n66	277593	277593	28	0.0013
12	56-tiw56n62	217499	217499	7	0.0008
13	56-tiw56n58	154440	154440	2	0.0015
14	56-tiw56n54	112767	112757	0	0.0013
15	50-be75tot	1127387	1127347	0	0.0012
16	50-be75oi	118159	118158	0	0.00004
17	50-be75np	790966	790963	0	0.000004
18	50-be75eec	264940	264638	0	0.0016
19	44-t75u1xx	63278034	63278034	13	0.0008
20	44-t75n11xx	113808	113808	77	0.0004
21	44-t75k11xx	124887	124887	63	0.0002

No.	Instances	Optimal	NACS-IM		
			Max.	No. Opt.	Avg, Dev.
22	44-t75i1xx	72664466	72664466	13	0.0005
23	44-t75e11xx	3095130	3095130	6	0.0021
24	44-t75d11xx	688601	688601	15	0.0059
25	44-t74d11xx	673346	673346	15	0.0024
26	44-t70x11xx	343471236	343471236	65	0.0002
27	44-t70w11xx	267807180	267807180	47	0.0004
28	44-t70u11xx	27296800	27296800	4	0.0007
29	44-t70n11xx	63944	63944	11	0.001
30	44-t70l11xx	28108	28108	6	0.0024
31	44-t70k11xx	69796200	69796200	18	0.0005
32	44-t70i11xx	28267738	28267738	3	0.0015
33	44-t70f11xx	413948	413948	14	0.0016
34	44-t70d11xx	450774	450774	36	0.0013
35	44-t70d11xn	438235	438235	14	0.0005
36	44-t70b11xx	623411	623411	47	0.0003
37	44-t69r11xx	865650	865650	2	0.0041
38	44-t65w11xx	166052789	166052789	2	0.0014
39	44-t65n11xx	38814	38814	22	0.0016
40	44-t65l11xx	18359	18359	81	0.0003
41	44-t65i11xx	16389651	16389651	5	0.0019
42	44-t65f11xx	254568	254515	0	0.0014
43	44-t65d11xx	283971	283969	0	0.0018
44	44-t65b11xx	411733	411733	2	0.0009
45	44-t59n11xx	25225	25225	57	0.0019

Table 1: NACS-IM results for LOLIB [12] obtained for five runs with 50000 iterations

The percentage average deviation of the obtained solution from the real optimal solution for all 45 LOLIB instances is 0.145 clearly outperforming recent evolutionary models [17] which report an average deviation of 0.714 for the same library. The results obtained by NACS-IM for LOLIB are further compared to local search methods [5] in terms of solution quality. Furthermore, instances from the SGB library are addressed using NACS-IM and the results are compared to variable neighborhood search [5]. Table 2 presents the comparative results (100 runs with 50000 iterations are considered for NACS-IM) indicating a similar performance of NACS-IM relative to state-of-the-art techniques – VNS. These are promising preliminary results and we expect their improvement (using other hybridizations) during further development.

Instances	Optimal	NACS-IM	
		Max.	Avg, Dev.
sgb75.1	6144679	6144646	0.0004
sgb75.2	6100491	6100377	0.0003
sgb75.3	6165775	6165474	0.0005
sgb75.4	6154958	6154958	0.0005
sgb75.5	6141070	6141070	0.0006
sgb75.6	6144055	6143861	0.0005
sgb75.7	6142899	6142831	0.0008
sgb75.8	6154094	6154062	0.0006
sgb75.9	6135459	6135459	0.0004

Instances	Optimal	NACS-IM	
		Max.	Avg, Dev.
sgb75.10	6149271	6149220	0.0006
sgb75.11	6151750	6151722	0.0005
sgb75.12	6150469	6150394	0.0009
sgb75.13	6156935	6156696	0.0005
sgb75.14	6149693	6149440	0.0009
sgb75.15	6150331	6150329	0.0004
sgb75.16	6164959	6164890	0.0006
sgb75.17	6163483	6163481	0.0005
sgb75.18	6063548	6062926	0.0004
sgb75.19	6150967	6150955	0.0002
sgb75.20	6152224	6152223	0.0013
sgb75.21	6159081	6159081	0.0005
sgb75.22	6127019	6127014	0.0002
sgb75.23	6136362	6135885	0.0005
sgb75.24	6168513	6166247	0.0008
sgb75.25	6150026	6149899	0.0004

Table 2: Numerical results for SGB instances obtained by NACS-IM compared to Variable Neighborhood Search (VNS) [5]

5. Conclusions and Future Work

The Triangulation Problem for Input-Output Matrices is solved with good results using a hybrid ant-based model. The proposed technique integrates an ACS model based on a greedy solution initialization and an Insert-Move mechanism used for local search. Numerical results obtained on some real-world economical data are encouraging for the potential of nature-inspired metaheuristics in solving linear ordering problems. NACS-IM is able to obtain higher-quality solutions to LOP compared to evolutionary techniques [17] and results comparable with state-of-theand neighborhood search methods [5]. Future work focuses on the investigation of other (more efficient) local search mechanisms to be considered for hybridization with the proposed ant-based model in order to improve the quality of LOP solutions. Additionally, it is planned to engage the proposed hybrid ant-based model for other available libraries of LOP instances to potentially show the benefits of using ant-based models when addressing higherdimensional problems.

References

[1] S. Chanas and P. Kobylanski, *A new heuristic algorithm solving the linear ordering problem*, Computational Optimization and Applications, 6(1996), 191-205.

[2] H. B. Chenery and T. Watanabe, *International comparisons of the structure of productions*, Econometrica, 26(4)(1958), 487-521.

[3] M. Dorigo and T. Stutzle, Ant Colony Optimization, MIT Press, Cambridge, (2004).

[4] M. Dorigo and L. M. Gambardella, *Ant Colony System: A cooperative learning approach in the traveling salesman problem*, IEEE Transactions on Evolutionary Computation, 1(1)(1997), 53-66.

[5] C. Garcia, D. Perez, V. Campos and R. Marti, *Variable neighborhood search for the linear ordering problem*, Computers and Operations Research, 33(2006), 3549–3565.

[6] F. Glover and M. Laguna, *Tabu Search*, Kluwer Academic Publisher, Boston, (1997).

[7] F. Glover, *A template for Scatter Search and Path Relinking*, In Hao, J.K., Lutton, E., Ronald, E., Schoenauer, M., Snyers, D. (eds.) Artificial Evolution. LNCS, 1363(1998), 13-54.

[8] R. Hassin and S. Rubinstein, *Approximations algorithms for maximum linear arrangement*, In: Halidorsson, M.M. (ed.) SWAT '00: Proceedings of the 7th Scandinavian Workshop on Algorithm Theory. LNCS, 1851, 231-236. Springer, London, (2000).

[9] R. Kaas, *A branch and bound algorithm for the acyclic subgraph problem*, European Journal of Operational Research, 8(1981), 355-362.

[10] D. E. Knuth, The Stanford GraphBase: A Platform for Combinatorial Computing, Addison Wesley, New York, (1993).

[11] H. R. Lourenco, O. Martin and T. Stutzle, *Handbook of Metaheuristics*, International Series in Operations Research & Management Science(2002), 57, 321-353.

[12] LOLIB site, http://www.iwr.uni-heidelberg.de/groups/comopt/software/LOLIB/

[13] F. L. Pryor, *Economic evolution and structure: The impact of complexity on the US economic system*, Cambridge University Press, (1996).

[14] J. E. Mitchell and B. Borchers, *Solving linear ordering problem with a combined interior point/simplex cutting plane algorithm*, High Performance Optimization, Kluwer Academic Publishers, (2000), 349-366.

[15] S. Mishra and K. Sikdar, *On approximability of linear ordering and related NP-optimization problems in graphs*, Discrete Applied Mathematics, 136(2-3)(2004), 249-269.

[16] G. Reinelt, *The linear ordering problem: algorithms and applications*, Research and Exposition in Mathematics, 8. Heldermann Verlag, Berlin, (1985).

[17] V. Snasel, P. Kromer and J. Platos, *Evolutionary approaches to linear ordering problem in DEXA Workshops*, IEEE Computer Society, (2008), 566–570.