American Journal of Operations Research, 2018, 8, 281-293
http://www.scirp.org/journal/ajor
ISSN Online: 2160-8849
ISSN Print: 2160-8830

Efficient Heuristic Based Methods for Two-Stage Transshipment Problem

Priyank Sinha, Renduchintala Raghavendra Kumar Sharma

Industrial and Management Engineering Department, IIT Kanpur, Kanpur, India
Email: spriyank@iitk.ac.in, rrks@iitk.ac.in

How to cite this paper: Sinha, P. and Sharma, R.R.K. (2018) Efficient Heuristic Based Methods for Two-Stage Transshipment Problem. *American Journal of Operations Research*, **8**, 281-293.
https://doi.org/10.4236/ajor.2018.84016

Received: May 21, 2018
Accepted: June 30, 2018
Published: July 3, 2018

Abstract

In this article, we propose efficient methods for solving two stage transshipment problems. Transshipment problem is the special case of Minimum cost flow problem in which arc capacities are infinite. We start by proposing a novel problem formulation for a two stage transshipment problem. Later, special structure of our problem formulation is utilized to devise two dual based heuristics solutions with computational complexity of O (n^2), and O (n^3) respectively. These methods are motivated by the methods developed by Sharma and Saxena [1], Sinha and Sharma [2]. Our methods differ in the initialization and the subsequent variation of the dual variables associated with the transshipment nodes along the shortest path. Lastly, a method is proposed to extract a very good primal solution from the given dual solutions with a computational complexity of O (n^2). Efficacy of these methods is demonstrated by our numerical analysis on 200 random problems.

Keywords

Two Stage Transshipment Problem, Min Cost Flow, Transportation Problem, Dual, Primal

1. Introduction

Minimum cost flow problem can be described as a preparation of minimum cost plan for the transportation of certain number of units from source to sink to satisfy the demand at each sink. One or more layers of transshipment nodes are present between the sources and sinks, which act as buffer. Weintraub [3] is attributed with the development of one of the first primal based algorithms to solve minimum cost flow problem with convex costs. A variant of negative cycle algorithm is used which aims to identify most negative cycle and subsequently induces it iteratively into the feasible flow. Plotkin and Tardos [4] later proposed a polynomial time algorithm for min cost flow with computational complexity of O (m^4). Pseudo flow is maintained by the algorithm such that it satisfies the complimentary slackness conditions while performing flow augmentation from the nodes with positive excess to nodes with negative excess. Algorithm is terminated when all the excesses are reduced to zero. Ahuja *et al.* [5] has employed advanced data structures to solve primal simplex algorithm for min-cost flow. They have further documented that transshipment problem can be solved by enhanced capacity scaling algorithm in O (n log (n) S (n, m)) time, where n and m are the number of nodes and number of arcs in a network respectively. S (n, m) refers to running time of shortest path problem with n nodes and m arcs. Enhanced capacity scaling algorithm and repeated capacity scaling algorithm runs in O (n^3 log (n)) and O (n^2 log (n) S (n, m)) time respectively (Ahuja *et al.* [5]). Further cost scaling algorithm can be used to solve min cost flow problem in O (n^3 log (n)). Hence, best primal based approach solves the transshipment problem with the time complexity of O (n^3 log (n)). Recently Juman and Hoque [6] have proposed an efficient heuristic with O (n^3) running time to solve uncapacitated balanced transportation problem. Performance of this heuristic is better than Vogel's Approximation Method (VAM), which is one of the most popular primal based methods to obtain a very good initial basic feasible solution.

Next, we review some popular dual based procedures to solve min cost flow problem. Busakar and Gowan [7] are attributed for developing successive shortest path algorithm. It successively identifies the least cost paths between the source and sink and allocates the maximum capacity on the residual network while maintaining the dual feasibility. Algorithm terminates when residual network contains no admissible path between the source and the sink or when the flow is maximal (primal feasibility is achieved). First polynomial time algorithm to solve min cost flow problem was proposed by Edmonds and Karp [8]. It is a specific implementation of Ford-Fulkerson algorithm in which breadth first search is used to find the shortest path in intermediate stages. Computational complexity of the algorithm is O ((n + m) log U). Helgason and Kennington [9] has used linear programming formulation of the network flow problem to analyze dual simplex method. Plotkin and Tardos [4] had proposed an efficient pivoting strategy with complexity of O (m^2n) for the dual simplex method proposed by Orlin [10]. This reduces the number of pivoting steps required for dual simplex method. According to Ali *et al.* [11], efficient execution of each pivot in dual based methods requires less number of iterations when compared with primal based algorithm. However, each pivot operation may require higher computational effort.

Our approach in this article to solve two stage transshipment problems is different from the approaches discussed above. We pose the two stage transshipment problem differently and exploit the special structure of the problem to devise two computationally attractive dual based procedures with the computational complexity of O (n^2) and O (n^3) respectively. We also propose a method (with computational complexity of O (n^2)) to extract good primal solution from an existing dual solution. These methods are motivated by Sharma and Saxena [1], and Sinha and Sharma [2]. Solution proposed by these methods can be used as an initial basic feasible solution to get an advanced start for the exact methods proposed by Orlin [10], Plotkin and Tardos [4], and Ali *et al.* [11]. Next we give problem formulation for two stage transshipment problem.

2. Problem Formulation

We next present the mathematical formulation of the primal problem and dual problem respectively.

2.1. Constants of the Problem

D_l refers to the absolute demand at the l^{th} demand node, while d_l is the demand at market l as a fraction of total market demand. Hence we have

$$d_l = D_l \Big/ \left[\sum_{l=1}^{L} D_l \right] \text{ and } \sum_{l=1}^{L} d_l = 1,$$

where L is the total number of demand nodes. Similarly S_i refers to to absolute number of units available for transportation at the source node i and $s_i = S_i \Big/ \left[\sum_{l=1}^{L} D_l \right]$. If the problem is balanced, then we have $\sum_{i=1}^{I} s_i = \sum_{l=1}^{L} d_l$, I is the total number of supply nodes. $C1_{ij}, C2_{jk}, C3_{kl}$ is the cost of transporting $\sum_{l=1}^{L} D_l$ units from node i to j, j to k and k to l respectively.

2.2. Decision Variables

$x1_{ij}, x2_{jk}, x3_{kl}$ is the absolute number of units transported from node i to j, j to k and k to l respectively. We further have

$$X1_{ij} = x1_{ij} \Big/ \left[\sum_{l=1}^{L} D_l \right], X2_{jk} = x2_{jk} \Big/ \left[\sum_{l=1}^{L} D_l \right], X3_{kl} = x3_{kl} \Big/ \left[\sum_{l=1}^{L} D_l \right]$$

Primal of the problem can be formulated as under –

2.3. Primal (P)

Minimize:

$$\sum_{i=1}^{I} \sum_{j=1}^{J} C1_{ij} X1_{ij} + \sum_{j=1}^{J} \sum_{k=1}^{K} C2_{jk} X2_{jk} + \sum_{k=1}^{K} \sum_{l=1}^{L} C3_{kl} X3_{kl} \quad (1)$$

Subject to:

$$\sum_{i=1}^{I} \sum_{j=1}^{J} X1_{ij} = 1 \quad \forall i \in I, j \in J \quad (2)$$

$$\sum_{j=1}^{J} \sum_{k=1}^{K} X2_{jk} = 1 \quad \forall j \in J, k \in K \quad (3)$$

$$\sum_{k=1}^{K} \sum_{l=1}^{L} X3_{kl} = 1 \quad \forall k \in K, l \in L \quad (4)$$

$$-\sum_{j;(i,j)\in N} X1_{ij} \geq -s_i \quad \forall i \in I \tag{5}$$

$$-\sum_{k;(k,l)\in N} X3_{kl} \geq -d_l \quad \forall l \in L \tag{6}$$

$$\sum_{k;(j,k)\in N} X2_{jk} - \sum_{i;(i,j)\in N} X1_{ij} = 0 \quad \forall j \in J \tag{7}$$

$$\sum_{l;(k,l)\in N} X3_{kl} - \sum_{j;(j,k)\in N} X2_{jk} = 0 \quad \forall k \in K \tag{8}$$

$$X1_{ij}, X2_{jk}, X3_{kl} \geq 0 \quad \forall i \in I, j \in J, k \in K \text{ and } l \in L \tag{9}$$

In this formulation we assume forward unidirectional flows. Equation (1) minimizes the cost of transportation of units from source nodes to sink nodes while satisfying the supply and demand constraints. Equation (2) ensures that entire supply is transported from supply nodes to meet the demand, which is valid for the balanced problem. Equation (3) and Equation (4) ensures that the entire supply is transported between the layers of transshipment nodes and demand nodes respectively. Equation (5) and Equation (6) are supply and demand constraints respectively. Equation (7) and equation (8) ensures that no inventory is accumulated on any transshipment nodes. Equation (9) is the non-negativity constraint.

2.4. Dual of the Problem (DP)

In this section we present the dual of the problem P. We associate $V1, V2, V3, U_i, V_l, W1_j, W2_k$ as the dual variables corresponding to (2), (3), (4), (5), (6), (7) and (8) respectively. We first state the dual of the problem as DP and then divide it into two parts as DP1 and DP2 for computational simplicity.

1) DP

Maximize:

$$V1 + V2 + V3 - \sum_{i\in I} s_i U_i - \sum_{l\in L} d_l V_l \tag{10}$$

Subject to:

$$V1 - U_i - W1_j \leq C1_{ij} \quad \forall i \in I, j \in J \tag{11}$$

$$V2 + W1_j - W2_k \leq C2_{jk} \quad \forall j \in J, k \in K \tag{12}$$

$$V3 - V_l + W2_k \leq C3_{kl} \quad \forall k \in K, l \in L \tag{13}$$

$$U_i, V_l \geq 0, \quad V1, V2, V3, W1_j, W2_k \quad \text{unrestricted} \tag{14}$$

2) DP1

Maximize:

$$V1 + V2 - \sum_{i\in I} s_i U_i \tag{15}$$

Subject to:

$$V1 - U_i - W1_j \leq C1_{ij} \quad \forall i \in I, j \in J \tag{16}$$

$$V2 + W1_j - W2_k \leq C2_{jk} \quad \forall j \in J, k \in K \tag{17}$$

$$U_i \geq 0,\ \ V1, V2, W1_j, W2_k \ \text{ unrestricted} \tag{18}$$

3) DP2

Maximize:

$$V3 - \sum_{l \in L} d_l V_l \tag{19}$$

Subject to:

$$V3 - V_l + W2_k \leq C3_{kl} \quad \forall k \in K, l \in L \tag{20}$$

$$V_l \geq 0,\ \ V3, W2_k \ \text{ unrestricted} \tag{21}$$

3. Theoretical Results

We start with the development of the heuristic for the dual solution, and then move on to develop heuristic for the primal. Let SP_{il} denote the length of shortest path from *i* to *l* such that $SP_{il} = \lambda_{ij} + \lambda_{jk} + \lambda_{kl}$, such that $\lambda_{ij}, \lambda_{jk}, \lambda_{kl}$ denotes the length of path from *i* to *j*, *j* to *k* and *k* to *l* in a shortest path SP_{il}.

SPS = { SP_{il} : SP_{il} is the shortest path between *i* and *l*} $\forall i \in I, l \in L$

Problem TP

Minimize:

$$\sum_{i,l} X_{il} SP_{il} \tag{22}$$

Subject to:

$$\sum_{l \in L} X_{il} = s_i \quad \forall i \in I \tag{23}$$

$$\sum_{i \in I} X_{il} = d_l \quad \forall l \in L \tag{24}$$

$$X_{il} \geq 0 \quad \forall i \in I, l \in L \tag{25}$$

Theorem 1: Problem P and Problem TP are equivalent.

Proof: Since all the arcs capacities are unbounded in the transshipment problem, hence optimal flow for a pair of source and sink nodes will always be on SP_{il}. This ensures that any further reduction in the objective function value is not possible. Therefore optimal solution to problem TP gives the optimal solution to problem P and hence problem TP and P are equivalent. **Hence proved**.

4. Solution Procedures

4.1. Description of Heuristic 1 (H1)

We extend this heuristic from our previous article on single stage transshipment Sinha and Sharma [2] problem to two stage transshipment problem. Unlike heuristic developed for single stage transshipment problem, we will decrease the $W1_j$ along the shortest path after substituting $U_i = 0\ \forall i \in I$. We next describe this heuristic in detail.

Step 1: Sort $SP_{il}\ \forall i \in I, j \in J$ in the ascending order and select the SP_{il} from top of the list.

Step 2: Assign $V1=\lambda_{ij}, V2=\lambda_{jk}, V3=\lambda_{kl}$, and $W1_j=\max_j\left(V1-C1_{ij}\right)\forall j\in J$

Step 3: $W2_k=\max_j\left(V2+W1_j-C2_{jk}\right)\forall k\in K$

$U_i=\max_j\left(V1-W1_j-C1_{ij}\right)\forall i\in I$

$V_l=\max_k\left(V3+W2_k-C3_{kl}\right)\forall l\in L$

Step 4: Evaluate DP = DP1 + DP2 and retain the best solution.

Step 5: Repeat step 3 by decreasing value of $W1_j$ in steps $\forall j\in J$.

Step 6: Goto step 1 and repeat the entire process for all SP_{il} from the sorted list.

Result 1: Computational complexity of the algorithm is O (n^2).

Proof: Complexity of this algorithm is dominated by the step 1 which can be solved in O (n^2) time.

4.2. Description of Heuristic 2 (H2)

In the previous algorithm, we tinkered with value of $W1_j$ along SP_{il} $\forall l\in L$. There is no reason as to why we cannot tinker with the values of $U_i, V_k, W2_k$ along SP_{il}. We define $\left(C1_{ij}-V1+U_i+W1_j\right)$, $\left(C2_{jk}-V2+W2_k-W1_j\right)$ and $\left(C3_{kl}-V3+V_l-W2_k\right)$ as source slack, transshipment slack and sink slack respectively ($S1, S2, S3$). Next we describe this heuristic in detail.

Step 0: Sort values of SP_{il} in SPS in ascending order and set $SP_{il}^*=\left(SP_{il}\right)_{\min}$, $j=1$.

Step 1: Set $W1_j=\max_j\left(V1-C1_{ij}\right), V1=\lambda_{ij}^*, V2=\lambda_{jk}^*, V3=\lambda_{kl}^*$ such that $\lambda_{ij}^*, \lambda_{jk}^*, \lambda_{kl}^*$ belongs to SP_{il}^* $\forall j\in J$.

Step 2: Compute max_value of DP1 and DP2 by increasing value of $U_i, V_l, W2_k$ and retain the maximum value of (DP1 + DP2).

Step 3: Decrease the value of $W1_j$ in steps and repeat the step 2.

Step 4: $j=j+1$, If $j>J$, then goto step 5 else goto step 1.

Step 5: Stop

Result 2: Heuristic 2 runs in O (n^3) time.

Proof: Complexity of the heuristic is dominated by step 2 which can be completed in O (n^3) steps.

4.3. Description of Heuristic 2 (H2)

We refer Sharma and Prasad [12] for developing a heuristic to get a good primal solution after having yielded a good dual solution. This is due to the structural similarity between the two problems. Solution to dual problem DP can be expressed as $DP, V1, V2, V3, U=\left\{U_i\ \forall i\in\left(1,I\right)\right\}, V=\left\{V_l\ \forall l=\left(i,L\right)\right\}$. Slack $S1, S2, S3$ is defined as following:

$S1=\left|C1_{ij}-V1+U_i+W1_j\right|$, $S2=\left|C2_{jk}-V2-W1_j+W2_k\right|$, $S3=\left|C3_{kl}-V3+V_l-W2_k\right|$ $\forall i\in\left(1,I\right), j\in\left(1,J\right), k\in\left(1,K\right)$. If $S1=0, S2=0$ and $S3=0$, then $X1_{ij}\ge 0$, $X2_{jk}\ge 0$ add $X3_{kl}\ge 0$. In other words primal variables for the arc $\left(i,j\right),\left(j,k\right),\left(k,l\right)$ (*i.e.* $X1_{ij}, X2_{jk}, X3_{kl}$) can assume positive value if the corresponding dual slacks for these arcs are zero. This is known as complementary slackness condition. We utilize this condition to devise a method to extract good primal solution from an existing dual solution. Let SP_{il} denote the total slack for a shortest path between node *i* and node *l* *i.e.*

$$SP_{il} = \min_{j,k}\left(\sum_{i,j} S1 + \sum_{j,k} S2 + \sum_{k,l} S3\right).$$

We further define deviation number as $DN_{il} = SP_{il} * X_{il}$. According to complimentary slackness condition, for a primal optimal solution, we have

$$\sum_{i\in I}\sum_{l\in L} DN_{il} = 0 .$$

If $SP_{il} = 0$, positive flow can be allocated to the corresponding path without violating the complimentary slackness property. Further, positive flow cannot be allocated to the path with $SP_{il} > 0$. As we are working with good dual solution (and not optimal dual solution), we may have to send positive flow along a path even if corresponding $SP_{il} > 0$. But the heuristic described below, tries to minimize DN_{il} and hope to keep complementary slackness violations as low as possible to get good primal solution. If algorithm terminates with

$$\sum_{i\in I}\sum_{l\in L} DN_{il} = 0 ,$$

then we have the optimal primal solution. DN_{il} is similar to Kilter number (Clasen [13]) for solving general min-cost-flow problem.

We find shortest path from every source node "*i*" to every sink node "*l*" using these slacks as weights, and then make the allocations according to shortest path available. Detailed heuristic is described as under:

Step 0: $X_{ij} = X_{jk} = X_{kl} = 0, \forall i \in I, j \in J, k \in K, l \in L$

Step 1: Compute $S_{il} = \min_{j,k}\left[S1_{ij} + S2_{jk} + S3_{kl}\right]$. Let min be at j^* and k^* $\forall i \in I, j \in J, k \in K, l \in L$.

$$S1_{ij} = \left|C1_{ij} - V1 + U_i + W1_j\right|,$$
$$S2_{jk} = \left|C2_{jk} - V2 - W1_j + W2_k\right|,$$
$$S3_{kl} = \left|C3_{kl} - V3 + V_l - W3_k\right|$$

Step 2: Sort all S_{il} in increasing order.

Step 3: Choose S_{il} from the top of the list such that $d_l > 0$ and $s_i > 0$ and goto step 4, if all S_{il} are processed then goto step 5

Step 4: $X_{ij^*} = X_{j^*k^*} = X_{k^*l} = \min(s_i, d_l) = a^*, s_i = s_i - a^*, d_l = d_l - a^*$, remove S_{il} from the list then goto step 3.

Step 5: Stop.

Result 3: Computational complexity of the algorithm is O (n^2).

Proof: Complexity of this algorithm is dominated by the step 2 which can be solved in O (n^2) time.

5. Numerical Analysis and Conclusions

In this section, we discuss the efficacy of our methods by implementing them on large size randomly generated problems. 200 problems of the size 200 × 200 ×

Table 1. Numerical analysis.

S. No.	Optimal Solution		Heuristic Solutions							
			H1			H2			H3	
	Value	No. of Iterations	Value	% closeness	No. of iterations	Value	% closeness	No. of iterations	Value	% closeness
1	25.603	1902	24.724	96.57	915	24.841	97.02	1093	26.002	98.44
2	28.389	1686	27.152	95.64	951	27.696	97.56	1136	29.069	97.6
3	23.348	1996	22.579	96.71	989	22.923	98.18	1004	23.766	98.21
4	26.266	1928	25.285	96.27	972	25.968	98.87	1007	27.448	95.5
5	29.317	1602	28.05	95.68	1085	28.479	97.14	1405	30.387	96.35
6	20.429	1897	19.761	96.73	1001	20.046	98.13	1347	20.963	97.39
7	22.95	2070	22.234	96.88	932	22.425	97.71	1054	22.96	99.96
8	24.807	1899	23.992	96.71	908	24.557	98.99	1118	25.76	96.16
9	25.338	1622	24.373	96.19	1010	24.831	98	1139	26.013	97.34
10	27.062	1608	26.063	96.31	992	26.321	97.26	1145	27.328	99.02
11	32.236	1656	31.357	97.27	1008	31.904	98.97	1108	32.354	99.63
12	22.552	1709	21.468	95.19	1035	22.321	98.98	1068	23.08	97.66
13	29.848	1910	29.139	97.62	936	29.201	97.83	1013	30.015	99.44
14	28.787	1868	27.936	97.04	923	28.486	98.95	1030	29.287	98.26
15	33.297	1723	32.311	97.04	1062	32.889	98.77	1195	33.566	99.19
16	27.195	1765	26.619	97.88	1041	26.752	98.37	1396	27.235	99.85
17	36.083	1600	35.041	97.11	1042	35.703	98.95	1400	37.425	96.28
18	38.338	1843	36.598	95.46	948	37.557	97.96	1110	38.432	99.75
19	32.103	1924	30.855	96.11	915	31.371	97.72	1220	33.341	96.14
20	32.501	1997	31.306	96.32	971	31.911	98.18	1217	32.937	98.66
21	32.103	1870	31.091	96.85	939	31.205	97.2	1139	32.999	97.21
22	28.521	1688	27.507	96.44	1038	28.031	98.28	1086	29.194	97.64
23	33.562	1803	31.92	95.11	964	32.767	97.63	1057	34.862	96.13
24	27.991	1921	26.914	96.15	1097	27.475	98.16	1268	28.573	97.92
25	26.399	1789	25.341	95.99	960	26.113	98.92	1257	27.291	96.62
26	30.113	1975	29.22	97.03	1001	29.441	97.77	1301	31.28	96.12
27	31.175	1780	30.248	97.03	910	30.427	97.6	1142	31.648	98.48
28	31.705	1717	30.929	97.55	1083	31.012	97.81	1225	32.16	98.56
29	28.521	1790	27.34	95.86	977	27.817	97.53	1224	28.958	98.47
30	32.103	1739	30.917	96.31	965	31.407	97.83	1230	33.207	96.56
31	31.573	2066	30.315	96.02	1030	30.738	97.36	1154	33.149	95.01
32	33.43	1744	32.073	95.94	1074	32.647	97.66	1191	33.92	98.53
33	36.348	1963	34.536	95.01	949	35.449	97.53	1178	37.078	97.99
34	28.919	1664	27.669	95.68	1076	28.479	98.48	1146	29.306	98.66
35	32.634	1804	31.533	96.63	982	32.021	98.12	1135	32.783	99.54
36	30.379	1772	29.458	96.97	1012	29.954	98.6	1118	30.674	99.03
37	31.573	1920	30.342	96.1	1049	30.833	97.66	1412	32.254	97.84
38	34.624	2033	33.096	95.59	983	34.175	98.7	1051	36.346	95.03
39	33.562	1687	32.627	97.21	1097	32.631	97.23	1210	34.597	96.92

Continued

40	32.766	2017	31.964	97.55	1048	32.378	98.82	1276	34.21	95.59
41	33.828	1795	32.903	97.27	968	33.474	98.95	1276	34.84	97.01
42	28.123	1984	27.533	97.9	1095	27.51	97.82	1220	28.862	97.37
43	28.256	1894	27.006	95.58	981	27.972	98.99	1238	29.427	95.86
44	30.777	1662	29.455	95.7	901	30.294	98.43	1047	31.647	97.17
45	27.46	1912	26.143	95.2	1091	26.995	98.31	1323	27.668	99.24
46	28.654	1868	27.282	95.21	977	28.074	97.98	1034	28.989	98.83
47	30.246	1766	29.132	96.32	1100	29.717	98.25	1325	31.417	96.13
48	27.593	1896	26.54	96.18	1084	26.841	97.27	1121	28.392	97.1
49	30.379	1808	29.537	97.23	928	29.877	98.35	1129	30.866	98.4
50	29.848	1945	29.079	97.42	977	29.197	97.82	1226	30.681	97.21
51	31.573	1750	30.496	96.59	1097	31.16	98.69	1099	33.006	95.46
52	29.317	1966	28.071	95.75	928	28.975	98.83	1008	30.075	97.41
53	29.848	1699	28.388	95.11	958	29.046	97.31	1237	30.094	99.18
54	30.644	1684	29.615	96.64	998	29.831	97.35	1263	30.885	99.21
55	28.919	1922	27.515	95.15	1001	28.42	98.27	1230	29.989	96.3
56	35.42	1885	33.791	95.4	1091	34.603	97.69	1223	36.64	96.56
57	33.562	1875	32.533	96.93	1077	33.152	98.78	1139	33.788	99.33
58	36.614	1853	34.998	95.59	945	35.967	98.23	1070	37.097	98.68
59	35.95	1939	34.515	96.01	1095	35.326	98.26	1305	36.183	99.35
60	33.032	1699	31.534	95.47	977	32.107	97.2	1148	33.21	99.46
61	33.828	1826	32.918	97.31	908	33.372	98.65	948	34.395	98.32
62	35.022	1858	34.046	97.21	904	34.557	98.67	922	36.04	97.09
63	32.634	1934	31.154	95.46	1030	32.224	98.74	1141	33.557	97.17
64	28.521	2089	27.782	97.41	982	27.785	97.42	1001	29.692	95.89
65	31.705	2081	30.801	97.15	1046	30.757	97.01	1240	33.151	95.44
66	30.777	1889	30.092	97.77	1091	30.078	97.73	1143	30.79	99.96
67	32.899	2050	31.784	96.61	942	32.164	97.77	1181	33.478	98.24
68	32.766	1835	31.181	95.16	1031	32.378	98.82	1065	33.969	96.33
69	29.981	1810	28.593	95.37	1033	29.453	98.24	1332	30.401	98.6
70	29.317	2072	28.279	96.46	1058	28.933	98.69	1317	30.329	96.55
71	28.654	1807	27.92	97.44	994	28.062	97.93	1156	29.984	95.36
72	30.246	1648	29.486	97.49	988	29.617	97.92	1017	31.496	95.87
73	27.593	1748	26.433	95.8	912	26.874	97.39	971	28.582	96.42
74	30.379	2011	29.242	96.26	1053	29.474	97.02	1171	31.868	95.1
75	29.848	1618	28.395	95.13	1022	29.125	97.58	1267	30.08	99.22
76	31.573	1775	30.619	96.98	973	30.689	97.2	1114	32.826	96.03
77	29.317	1911	27.907	95.19	1014	28.624	97.64	1310	29.478	99.45
78	29.848	1809	29.078	97.42	962	29.158	97.69	1056	30.743	97
79	30.644	1862	29.983	97.84	1029	30.033	98.01	1229	31.761	96.35
80	28.919	2056	27.679	95.71	1003	28.305	97.88	1058	28.994	99.74
81	35.42	1952	34.63	97.77	948	34.373	97.04	1079	36.863	95.93
82	33.562	1644	32.607	97.15	1030	32.67	97.34	1207	34.431	97.41

Continued

83	36.614	1635	35.325	96.48	989	35.862	97.95	1064	37.161	98.51
84	35.95	2083	34.376	95.62	1010	34.983	97.31	1112	35.959	99.97
85	33.032	1621	31.486	95.32	961	32.315	97.83	1034	34.665	95.06
86	33.828	1655	32.966	97.45	1078	33.328	98.52	1091	33.844	99.95
87	35.022	1739	33.572	95.86	1041	34.336	98.04	1161	35.794	97.8
88	32.634	1882	31.102	95.31	1073	32.127	98.45	1347	33.935	96.01
89	28.521	1833	27.607	96.8	997	27.671	97.02	1319	29.437	96.79
90	31.705	1908	30.177	95.18	1044	31.282	98.67	1179	32.171	98.53
91	28.654	2100	27.304	95.29	1090	28.216	98.47	1256	29.47	97.15
92	30.246	1623	29.174	96.46	964	29.584	97.81	972	31.334	96.4
93	27.593	1761	26.694	96.74	1070	26.936	97.62	1100	28.672	96.09
94	30.379	1732	29.673	97.68	1097	29.641	97.57	1216	31.743	95.51
95	29.848	1746	29.165	97.71	1092	29.038	97.29	1145	31.008	96.11
96	31.573	2093	30.271	95.88	933	30.749	97.39	1095	31.814	99.24
97	29.317	1792	28.009	95.54	1096	28.583	97.5	1293	30.008	97.64
98	29.848	1973	28.944	96.97	951	29.271	98.07	1072	30.638	97.35
99	30.644	1805	29.88	97.51	944	29.793	97.22	1176	31.521	97.14
100	28.919	2051	28.073	97.07	991	28.373	98.11	1250	29.095	99.39
101	35.42	1989	33.983	95.94	922	34.713	98	968	36.873	95.9
102	33.562	1635	32.64	97.25	1074	32.678	97.37	1321	34.599	96.91
103	36.614	1637	35.487	96.92	967	35.72	97.56	973	38.221	95.61
104	35.95	1847	35.098	97.63	991	35.24	98.03	1008	37.664	95.23
105	33.032	1672	32.005	96.89	1010	32.055	97.04	1158	34.67	95.04
106	33.828	1910	32.56	96.25	948	33.322	98.5	1233	34.991	96.56
107	35.022	1873	34.029	97.16	922	34.45	98.37	1070	36.456	95.91
108	32.634	1867	31.951	97.91	1022	31.71	97.17	1188	33.124	98.5
109	28.521	1639	27.166	95.25	904	27.937	97.95	1119	29.465	96.69
110	31.705	2062	30.197	95.24	919	31.28	98.66	1023	32.43	97.71
111	28.654	1648	27.823	97.1	1000	28.245	98.57	1099	29.432	97.28
112	30.246	1725	29.14	96.34	941	29.813	98.57	1075	31.376	96.26
113	27.593	1876	26.26	95.17	1098	27.149	98.39	1396	27.987	98.57
114	30.379	1845	29.487	97.06	1065	29.637	97.56	1272	30.767	98.72
115	29.848	1947	28.453	95.33	1096	29.219	97.89	1123	30.496	97.83
116	31.573	1670	30.896	97.86	986	30.726	97.32	995	31.97	98.74
117	29.317	1997	27.993	95.48	1024	28.873	98.49	1291	30.492	95.99
118	29.848	1902	28.961	97.03	991	28.976	97.08	1094	29.996	99.5
119	30.644	1707	29.963	97.78	965	30.195	98.53	1251	32.093	95.27
120	28.919	1714	27.974	96.73	1017	28.537	98.68	1167	30.3	95.22
121	35.42	1697	34.098	96.27	1078	34.58	97.63	1422	36.336	97.41
122	33.562	1672	32.465	96.73	1042	32.567	97.04	1361	33.799	99.29
123	36.614	1632	35.431	96.77	1064	36.16	98.76	1384	38.335	95.3
124	35.95	1945	34.944	97.2	1046	35.484	98.7	1271	35.982	99.91
125	33.032	1993	31.835	96.38	901	32.269	97.69	1204	33.681	98.04

Continued

126	33.828	1975	32.522	96.14	947	33.38	98.68	1039	34.969	96.63
127	35.022	1802	34.284	97.89	1029	34.039	97.19	1311	36.592	95.52
128	32.634	1710	31.804	97.46	978	31.755	97.31	1082	32.809	99.46
129	28.521	2036	27.511	96.46	923	27.68	97.05	1189	29.437	96.79
130	31.705	1945	30.32	95.63	907	31.305	98.74	1178	32.816	96.5
131	28.654	1709	27.329	95.38	1023	28.345	98.92	1208	29.311	97.71
132	30.246	1696	29.613	97.91	911	29.535	97.65	930	31.162	96.97
133	27.593	1789	27.014	97.9	926	27.091	98.18	1201	28.907	95.24
134	30.379	1688	29.253	96.29	997	29.672	97.67	1128	31.763	95.44
135	35.42	1938	34.468	97.31	909	34.359	97	1183	35.656	99.33
136	33.562	1783	32.864	97.92	927	32.785	97.68	1168	34.262	97.91
137	36.614	1872	35.053	95.74	1082	35.981	98.27	1432	38.223	95.61
138	35.95	1661	35.187	97.88	931	34.965	97.26	990	37.1	96.8
139	33.032	1874	31.656	95.83	1011	32.544	98.52	1330	34.014	97.03
140	33.828	1728	32.797	96.95	1078	33.337	98.55	1246	34.237	98.79
141	35.022	1825	33.467	95.56	1043	34.125	97.44	1160	35.261	99.32
142	32.634	2086	31.325	95.99	919	31.877	97.68	1067	34.034	95.71
143	28.521	1647	27.504	96.43	1077	27.907	97.85	1090	28.917	98.61
144	31.705	2018	30.83	97.24	1062	30.881	97.4	1145	32.629	97.09
145	30.777	1999	29.414	95.57	1094	30.424	98.85	1380	31.287	98.34
146	32.899	1759	32.042	97.4	1062	32.172	97.79	1113	33.638	97.75
147	32.766	1692	31.13	95.01	966	32.031	97.76	1027	32.906	99.57
148	29.981	1690	28.568	95.29	904	29.574	98.64	907	30.3	98.94
149	29.317	1939	28.522	97.29	1030	28.654	97.74	1038	29.728	98.6
150	28.654	1720	27.827	97.11	1012	28.286	98.72	1307	29.607	96.67
151	31.811	1632	31.137	97.88	1087	31.035	97.56	1386	32.632	97.42
152	29.53	1666	28.483	96.45	1083	28.911	97.9	1441	29.802	99.08
153	30.074	1800	29.327	97.52	1053	29.246	97.25	1406	30.424	98.84
154	30.869	2084	29.676	96.14	951	30.324	98.23	1264	32.205	95.67
155	29.132	1762	28.216	96.86	1046	28.46	97.69	1063	29.344	99.27
156	35.685	1963	34.605	96.97	916	35.129	98.44	1013	37.414	95.15
157	33.814	1725	32.64	96.53	904	33.176	98.11	1198	34.084	99.2
158	36.879	1774	35.697	96.79	922	36.113	97.92	1214	38.068	96.78
159	36.216	1639	35.334	97.56	986	35.306	97.49	1135	37.799	95.63
160	33.271	1907	31.929	95.97	928	32.537	97.79	989	34.617	95.95
161	34.08	1896	32.697	95.94	928	33.367	97.91	973	35.279	96.48
162	35.274	1782	34.415	97.56	926	34.365	97.42	1191	35.909	98.2
163	32.873	1948	32.085	97.6	1070	31.967	97.24	1100	34.001	96.57
164	28.734	1796	27.386	95.31	1032	28.01	97.48	1064	29.12	98.66
165	31.944	1875	30.738	96.22	914	31.341	98.11	1165	33.383	95.5
166	31.002	1632	30.117	97.15	917	30.22	97.48	1179	31.638	97.95
167	33.138	1760	32.371	97.69	1093	32.459	97.95	1122	34.161	96.91
168	33.005	1679	32.263	97.75	1051	32.343	97.99	1085	33.857	97.42

Continued

169	30.206	1690	29.525	97.75	924	29.786	98.61	1148	30.683	98.42
170	29.53	1904	28.175	95.41	983	29.142	98.69	1251	30.241	97.59
171	28.866	2099	27.465	95.15	1022	28.497	98.72	1063	29.962	96.2
172	30.471	1844	29.125	95.58	1013	30.116	98.83	1152	30.899	98.6
173	27.792	1925	26.911	96.83	1075	27.284	98.17	1288	28.518	97.39
174	30.604	1613	29.843	97.51	1046	30.265	98.89	1169	31.232	97.95
175	30.074	1796	29.029	96.53	916	29.263	97.3	1195	30.437	98.79
176	31.811	1868	30.529	95.97	954	31.036	97.56	1080	32.345	98.32
177	29.53	1925	28.805	97.54	991	29.178	98.81	1041	30.377	97.13
178	30.074	2083	28.801	95.77	1066	29.306	97.45	1235	31.112	96.55
179	30.869	1919	29.589	95.85	927	30	97.18	952	31.952	96.49
180	29.132	2088	28.527	97.92	943	28.324	97.23	990	30.496	95.32
181	35.685	1690	34.545	96.81	1012	35.057	98.24	1062	37.088	96.07
182	33.814	2034	32.961	97.48	1019	33.378	98.71	1128	34.447	98.13
183	36.879	2022	35.037	95.01	1095	36.445	98.82	1417	37.817	97.46
184	36.216	1620	34.491	95.24	1032	35.826	98.92	1243	37.65	96.04
185	33.271	2002	32.55	97.83	1030	32.451	97.54	1079	33.854	98.25
186	34.08	1758	32.565	95.55	1074	33.42	98.06	1393	35.392	96.15
187	35.274	1891	33.892	96.08	1080	34.277	97.17	1323	36.225	97.3
188	32.873	2086	31.872	96.95	928	32.105	97.66	1086	33.52	98.03
189	28.734	2032	28.011	97.48	934	28.154	97.98	1126	28.784	99.83
190	31.944	1978	30.427	95.25	1049	31.009	97.07	1201	32.8	97.32
191	28.866	2012	28.272	97.94	968	28.319	98.11	1230	29.195	98.86
192	30.471	1810	29.861	98	1046	30.158	98.97	1249	31.487	96.67
193	27.792	2088	27.171	97.77	988	27.048	97.32	1078	27.85	99.79
194	30.604	1643	29.799	97.37	1083	29.753	97.22	1324	30.702	99.68
195	30.074	2005	28.642	95.24	977	29.371	97.66	1075	30.22	99.51
196	31.811	1891	31.098	97.76	967	31.208	98.1	1145	32.195	98.79
197	29.53	1692	28.792	97.5	1050	29.006	98.23	1300	30.518	96.65
198	30.074	1786	28.779	95.69	975	29.507	98.11	1033	31.367	95.7
199	30.869	1981	30.046	97.33	1034	30.071	97.41	1394	30.943	99.76
200	29.132	1879	28.038	96.24	998	28.819	98.93	1206	29.394	99.1

200 × 200 are solved. Heuristic methods on these problems are implemented in C++. Optimal solution was obtained from problem implementation in CPLEX in GAMS. We compare the number of iterations and the duality gap for these problems in **Table 1**. As shown, our methods are capable of reducing the computational effort (number of iterations) significantly while not compromising much on duality gap. Heuristic solutions by H1 and H2 are capable of achieving on an average 96% and 98% closeness to optimal solution. Computational complexity of these methods has already been discussed. We later intend to extend this work to general case of Minimum cost flow problem with finite arc capacities.

References

[1] Sharma, R.R.K. and Saxena, A. (2002) Dual Based Procedures for the Special Case of Transshipment Problem. *Operation Research*, **39**, 177-188.

[2] Sinha, P. and Sharma, R.R.K. (2016) Dual Based Procedures for Un-Capacitated Minimum Cost Flow Problem. *American Journal of Operations Research*, **6**, 468-479. https://doi.org/10.4236/ajor.2016.66043

[3] Weintraub, A. (1974) A Primal Algorithm to Solve Network Flow Problems with Convex Costs. *Management Science*, **21**, 87-97. https://doi.org/10.1287/mnsc.21.1.87

[4] Plotkin, S.A. and Tardos, E. (1990) Improved Dual Network Simplex. *Proceedings of the* 1*st* *Annual ACM-SIAM Symposium on Discrete Algorithms*, *Society for Industrial and Applied Mathematics*, San Francisco, 367-376. http://delivery.acm.org/10.1145/330000/320222/p367-plotkin.pdf?ip=14.139.38.9&id=320222&acc=ACTIVE%20SERVICE&key=045416EF4DDA69D9%2E6454B2DFDB9CC807%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&__acm__=1526448927_7f6da1e85e11c5cff23e4cbeb14d250d

[5] Ahuja, R.K. (1993) Network Flows. Ph.D. Thesis, Technische Hochshule Darmstadt, Darmstadt. https://volyubemw.updog.co/dm9seXViZW13MDEzNjE3NTQ5WA.pdf

[6] Juman, Z.A.M.S. and Hoque, M.A. (2015) An Efficient Heuristic to Obtain a Better Initial Feasible Solution to the Transportation Problem. *Applied Soft Computing*, **34**, 813-826. https://doi.org/10.1016/j.asoc.2015.05.009

[7] Busaker, R.G. and Gowen, P.J. (1961) A Procedure for Determining Minimal-Cost Flow Network Patterns. Tech. Rep. ORO-15, Operational Research Office, Johns Hopkins University, Baltimore.

[8] Edmonds, J. and Karp, R.M. (1972) Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems. *Association for Computing Machinery Journal*, **19**, 248-264. https://doi.org/10.1145/321694.321699

[9] Helgason, R.V. and Kennington, J.L. (1977) An Efficient Procedure for Implementing a Dual Simplex Network Flow Algorithm. *AIIE Transactions*, **9**, 63-68. https://doi.org/10.1080/05695557708975122

[10] Orlin, J.B. (1984) Genuinely Polynomial Simplex and Non-Simplex Algorithms for Minimum Cost Problems. Technical Report 1615-84, Sloan School of Management, MIT, Cambridge. https://dspace.mit.edu/bitstream/handle/1721.1/48015/genuinelypolynom00orli.pdf?sequence=1

[11] Ali, A.I., Padman, R. and Thiagarajan, H. (1989) Dual Algorithms for Pure Network Problems. *Operations Research*, **37**, 159-171. https://doi.org/10.1287/opre.37.1.159

[12] Sharma, R.R.K. and Prasad, S. (2003) Obtaining a Good Primal Solution to the Uncapacitated Transportation Problem. *European Journal of Operational Research*, **144**, 560-564. https://doi.org/10.1016/S0377-2217(01)00396-4

[13] Clasen, R.J. (1968) The Numerical Solution of Network Problems Using the Out-of-Kilter Algorithm. No. RM-5456-PR. RAND CORP Santa Monica. http://www.dtic.mil/docs/citations/AD0667528