KM 14(1984)
pag 134-141

A Primal method for the Assignment Problem

J.N.M. van Loon *

Abstract

This paper describes a primal method for the assignment problem. The algorithm is based on a "Tomizawa-step", making at least one column of the relative-cost matrix dual feasible. So the total number of steps is less than or equal to n, the problem size.
The number of degenerate pivots is negligible. The advantages of a primal method are well-known: in each stage of the calculation a feasible assignment is available and the process can be started with a "good" primal solution. Computational experience is given.

* Koninklijke Militaire Academie
Breda, tel. 076-223911-2163

1. Introduction

We consider the linear assignment problem and its dual

$$
\begin{array}{llll}
\text{Min } \sum\limits_{ij}\sum a_{ij}x_{ij} & & \qquad \text{Max } \sum\limits_i u_i + \sum\limits_j v_j & \\
\text{s.t. } \sum\limits_j x_{ij} = 1 & i=1,\dots,n & \qquad \text{s.t. } u_i + v_j \leqslant a_{ij} & \\
\phantom{\text{s.t. }} \sum\limits_i x_{ij} = 1 & j=1,\dots,n & & i=1,\dots,n \\
 & & & j=1,\dots,n \\
\phantom{\text{s.t. }} x_{ij} \geqslant 0 & i=1,\dots,n \;;\; j=1,\dots,n & &
\end{array}
$$

The best known algorithm for solving this problem is perhaps the primal-dual algorithm of Tomizawa, revised by Dorhout [2]. Essential is the use of Dijkstra's shortest path method, as in the relaxation method of Hung and Rom [3]. Balinski and Gomory [1] introduced a primal method, with the advantage of having a complete feasible assignment in each step of the algorithm. Primal methods like the simplex method usually require many degenerate steps.

In [3] computational experience is mentioned, showing that also the modified simplex method of Balinski and Gomory is still suffering from it.

In the present method the number of degenerate steps, where the primal solution does not change, is negligible. In one step at least one column of the reduced-cost matrix is made dual feasible, where several degenerate steps in the method of Balinski and Gomory are needed.

2. A primal method

Suppose that a primal solution $\{x_{ij}\}$ with some corresponding dual solution $\{u_i, v_j\}$ is at hand, either found during the calculation or constructed in some other way, with at least one negative entry $\bar{a}_{ij} = a_{ij} - u_i - v_j$ of the corresponding reduced-cost matrix. So the dual solution is not feasible, that means the primal solution is not optimal.

Select a column j_o with some $a_{ij_o} < 0$ and choose i_o such that

$$\bar{a}_{i_oj_o} = \operatorname*{Min}_i \{\bar{a}_{ij_o} \mid \bar{a}_{ij_o} < 0\}$$

and make

$$v_{j_o} := v_{j_o} + a_{i_o j_o} \quad ^{1)} \quad \text{resp.} \quad \bar{a}_{ij_o} := \bar{a}_{ij_o} - \bar{a}_{i_o j_o} \geqslant 0 \quad \text{for } i=1,\ldots,n$$

Next we look for the best corresponding primal solution. Consider the momentary primal solution and cancel the assignment in column j_o and let us say row i_1. The shortest path from row i_1 to column j_o can be found by using a slightly revised "Tomizawa"-step. See steps 4, 5 and 6 of the revised algorithm in [2]. Essential for our method is keeping all nonnegative $\bar{a}_{ij}$ nonnegative. This can be achieved by using only the nonnegative entries of the reduced cost matrix. The result is a new complete assignment, with at least column j_o as an extra dual feasible column. Therefore the algorithm terminates in at most n steps. Degeneracy only occurs when element (i_1, j_o) gives the shortest path between row i_1 and column j_o.
The validity of the algorithm is on the one hand based on the well-known theorem that the optimal solution of the problem does not change when a constant is added to all elements of a row or column of the cost matrix. On the other hand we refer to the validation of the Tomizawa algorithm in [2]. Justification of the nonnegativity condition in this step, as applied in our method, is immediate.

3. The Algorithm

step 0. Initialization.

Start with a primal feasible solution $\{x_{ij}\}$

Define the labels $\xi_i = j$ and $\eta_j = i$ iff $x_{ij} = 1$

for $i = 1,\ldots,n$; $j = 1,\ldots,n$

Construct a corresponding dual solution, e.g.

$u_i = 0$ for $i = 1,\ldots,n$

$v_j = a_{\eta_j,j}$ for $j = 1,\ldots,n$

Reduce the cost matrix: $\bar{a}_{ij} := a_{ij} - u_i - v_j$ for $i = 1,\ldots,n$; $j = 1,\ldots,n$

step 1. Column selection.

Determine $\bar{a}_{i_o j_o} = \underset{i,j}{\text{Min}} \{\bar{a}_{ij} | \bar{a}_{ij} < 0\}$

The method stops if all $\bar{a}_{ij} \geqslant 0$: the solution is dual feasible, hence optimal.

1) a := b stands for 'a is replaced by b'

step 2. Preparation for Tomizawa-step:

Make $\Delta v_{j_o} := \bar{a}_{i_o j_o}$ and $\bar{a}_{ij_o} := \bar{a}_{ij_o} - \Delta v_{j_o}$ for $i = 1,\ldots,n$

$v_{j_o} := v_{j_o} + \bar{a}_{i_o j_o}$

Cancel the assignment in column j_o and row $\eta_{j_o} = i_1$: $\xi_{i_1} := 0$

$\eta_{j_o} := 0$

step 3. Tomizawa-step:

a. Define a set $T = \{1,\ldots,n\}$ and labels $\lambda_i = j_o$ for all $i \in T$

Make $\Delta u_i := \bar{a}_{ij_o}$ for $i = 1,\ldots,n$;

$\Delta v_{j_o} := 0$ and $\Delta v_j = \infty$ for all $j \neq j_o$

b. If $\Delta u_m = \underset{i}{\text{Min}} \{\Delta u_i | i \in T\}$ then go to c. in the case that

$\xi_m = 0$, $T := T - \{m\}$, $j := \xi_m$, $\Delta v_j := \Delta u_m$.

For all $i \in T$, with $\bar{a}_{ij} \geqslant 0$ and $\bar{a}_{ij} + \Delta v_j < \Delta u_i$,

make $\lambda_i := j$ and $\Delta u_i := \bar{a}_{ij} + \Delta v_j$. Repeat this step.

c. Construct the shortest path from column j_o to row m, using the labels λ_i . The dual variables become:

$u_i := u_i + \text{Min}(\Delta u_m , \Delta u_i) \quad i = 1,\ldots,n$

$v_j := v_j - \text{Min}(\Delta u_m , \Delta v_j) \quad j = 1,\ldots,n$

And the reduced matrix: $\bar{a}_{ij} := \bar{a}_{ij} - u_i - v_j$ for $i = 1,..,n;$
$j = 1,..,n$

After this step the assignment is complete again and at least column j_o is feasible. Meanwhile all other nonnegative elements $\bar{a}_{ij}$ remain feasible.

Go to step 1.

4. <u>Example</u>

Consider the assignment problem from [2], defined by the cost matrix (see fig. 1).

step 0. Start with $x_{ii} = 1$, $u_i = 0$, $v_i = a_{ii}$
assigned cells are encircled.

step 1. $\bar{a}_{i_o j_o} = \bar{a}_{54} = -8$

						u_i
	7	12	9	11	5	0
	5	10	7	8	12	0
	14	15	13	12	8	0
	8	13	11	14	7	0
	10	9	7	6	13	0
v_j	7	10	13	14	13	

order	T	λ_i						Δu_i
(3)	~~1~~	2~~4~~	0	2	-4	-~~3~~5	-8	~~5~~4
(2)	~~2~~	4	-2	0	-6	-~~6~~2	-1	2
	3	4	7	5	0	-~~2~~6	-5	~~6~~5
(4)	~~4~~	2~~4~~	1	3	-2	~~0~~8	-6	~~8~~5
(1)	~~5~~	4	3	-1	-6	-~~8~~0	0	0
						-8		
Δv_j			~~0~~	~~0~~	~~0~~	0	~~0~~	
			4	2	5		0	

order	λ_i						Δu_i
(1)	5	0	0	-3	1	-12	0
(2)	1~~5~~	0	0	-3	0	-3	~~9~~0
	4~~5~~	6	2	0	1	-10	~~2~~~~1~~0
(3)	1~~5~~	0	0	-2	3	-11	~~1~~0
(4)	4~~1~~~~5~~	7	1	-1	0	~~0~~	~~1~~~~2~~~~1~~0
						-12 [1)]	
Δv_j		~~0~~	~~0~~	~~0~~	~~0~~	0	
		0	0	0	0		

order	λ_i						Δu_i
(1)	3	0	0	-3	1	0	0
(2)	3	0	0	-3	0	9	0
(5)	5~~3~~	6	2	~~0~~	1	2	~~3~~2
(3)	1~~3~~	0	0	-2	3	1	~~1~~0
(4)	2~~3~~	7	1	-1	0	12	~~2~~1
				-3 [1)]			
Δv_j		~~0~~	~~0~~	0	~~0~~	~~0~~	
		0	0		1	0	

						u_i
	0	0	0	2	0	4
	0	0	0	1	9	2
	4	0	1	0	0	7
	0	0	1	4	1	5
	6	0	1	0	11	1
v_j	3	8	5	5	1	

fig. 1. Example.

1) Is not yet subtracted from the column.

step 2. $\Delta v_4 = -8$ is used in column 4, assignment (4,4) is cancelled

step 3a. Labels λ_i are mentioned in the left-hand margin, $\Delta v_4 = 0$.

step 3b. $\Delta u_m = \Delta u_5 = 0$, $j = 5$. The order of computation is given between brackets. $\Delta v_5 = 0$, no labels are changed by column 5. $\Delta u_m = \Delta u_2 = 2$, $j = 2$, $\Delta v_2 = 2$, change $\Delta u_1 = 4$, $\lambda_1 = 2$ and $\Delta u_4 = 5$, $\lambda_4 = 2$. $\Delta u_m = \Delta u_1 = 4$, $\Delta v_1 = 4$, no labels changed . $\Delta u_m = \Delta u_4 = 5$. Go to step 3c.

step 3c. $\Delta u_m = 5$ so $\Delta v_3 = 5$ and $\Delta u_3 = 5$
Column 4 is feasible, but also columns 1 and 2. Go to step 1. In the next iteration the cost matrix is not changed, but the primal solution is. After one more iteration the optimum is found. The number of iterations is 3. There are no degenerate steps.

5. Computational Experience

The algorithm is tested against the primal method of Balinski and Gomory [1] and the dual method of Tomizawa, improved by Dorhout [2] . The last one being the best method at present, at least in my knowledge. The first one was merely used as a starting point in my search for a primal method. Experience with this algorithm in [3] shows that more than 90% of the pivots in the problems tested are degenerate. In our method the number of degenerate pivots, these are iteration steps where the primal solution does not change, is negligible. Several degenerate steps needed to make a column of the reduced-cost matrix dual feasible are contracted to one step in the present method. Therefore in the computations we only compared with the Tomizawa method.
In order to get a fair comparison we programmed this method in a similar way as the proposed algorithm. The advantages of the present method, giving a primal feasible solution at each stage of the calculation and enabling to start with a known "good" primal solution, cannot be shown in computer results. However we can test average behaviour against the Tomizawa method.

Three initialization methods were used:

(I_1): $x_{ii} = 1$ for $i = 1,\ldots,n$ and $x_{ij} = 0$ for $i \neq j$

(I_2): $x_{ij} = 1$ if $a_{ij} = \operatorname{Min}_k \{a_{kj} | \text{no assignment in row } k\}$, successivily for $j = 1,\ldots,n$.

(I_3): After row- and columnreduction and straightforward assignment of independent zeroes (see [2]) use method I_2 for not-assigned rows and columns.

The algorithm was programmed in Basic and run on a VAX 11/750 time-sharing system. The test problems are randomly generated with cost coefficients between 1 and 99. The results shown in Table I come from fixed problems. The results in Table II are average results over each 5 randomly generated problems.

In order to test the sensitivity for variation in the range of cost coefficients we also generated problems with $1 \leq a_{ij} \leq 999$ and $40 \leq a_{ij} \leq 50$.

The results gave about the same tendencies.

Finally the selection of column j_o in step 1 of the algorithm was changed as follows:

Step 1: Select the first infeasible column j_o and determine

$$\bar{a}_{i_o j_o} = \operatorname{Min}_i \{\bar{a}_{ij_o} | \bar{a}_{ij_o} < 0\}$$

The results were slightly higher CPU-times, numbers of degenerate steps and iterations.

TABLE I

results on fixed (random) problems

	I_1				I_2				I_3				Tomizawa
n	n^+	deg	it	T	n^+	deg	it	T	n^+	deg	it	T	T
10	2	1	8	4	5	2	3	1	7	1	3	2	1
20	0	0	19	34	14	2	4	7	15	0	4	9	10
30	0	0	28	97	15	4	12	37	22	1	7	24	24
50	0	1	49	448	29	2	14	123	38	1	11	104	90
80	1	0	77	1720	50	2	20	453	65	1	15	337	271
100	1	1	98	3453	64	4	29	947	77	2	21	708	578

Remarks: $1 \leq a_{ij} \leq 99$, CPU-times in 0,1 secs.

TABLE II

Average Results on random problems

n	n^+	deg	it	T	Tomizawa T
10	8,0	0	2,0	2,0	1,5
20	17,2	0,4	4,0	8,4	6,3
30	24,0	1,0	5,8	25,2	20,3
50	38,8	1,2	10,0	103,0	69,9
80	64,6	1,4	13,8	336,6	300,6
100	79,8	1,0	18,4	704,0	536,0

Remarks:

5 problems in each problem size

$1 \leqslant a_{ij} \leqslant 99$

CPU-times in 0,1 secs

initialization I_3

n^+ = number of nonnegative columns in the reduced-cost matrix after initialization

deg= number of degenerate steps

it = total number of iterations

T = CPU-time

Concluding remarks: In the average and using initialization method I_3 it seems that: Total number of iterations $\cong$ 0,19.n

CPU-time "primal" $\cong$ 1,3."Tomizawa"

$n^+ \cong 0,8.n$

deg $\cong$ (0 - 0,1).n

References

[1] Balinski, M.L. and Gomory, R.E.: "A primal method for the assignment and transportation problems", Man. Sc.,Vol 10, No 3 (1964).

[2] Dorhout, B.: "Experiments with some algorithms for the linear assignment problem", Report BW 39/77, Mathematisch Centrum, Amsterdam (1977).

[3] Hung, M.S. and Rom, W.O.: "Solving the assignment problem by relaxation", Op. Res., Vol 28, No 4 (1980).

Ontvangen: 23-2-1984