Teaching linear assignment by Mack's algorithm

R. Jonker
Department of Mathematics and Systems Engineering
Koninklijke/Shell Laboratorium, Amsterdam

A. Volgenant
Institute of Actuarial Sciences and Econometrics
University of Amsterdam

In many introductory textbooks on Operations Research and Mathematical Programming the Linear Assignment Problem is solved by the Hungarian Method, probably because this is the oldest method. We highlight another method, the Bradford method of Mack; we think it to be more suited to explain a solution method for the Linear Assignment Problem. After a short review of assignment algorithms, we give an exposition of elements of these algorithms. Then we explain Mack's algorithm illustrated with an example and we will show it to be near - equivalent to the Hungarian Method. Finally we indicate an improvement on Mack's Method.

1. Introduction

In many introductory textbooks on Operations Research and Mathematical Programming, e.g., Daellenbach, George and McNickle (1983), and Taha (1987), the Linear Assignment Problem is introduced by the Hungarian Method, probably because this is the oldest method and maybe because not many textbooks treat any other method.

We highlight another method, the Bradford method of Mack (1969), which is theoretically equivalent to the Hungarian method; we think it to be more suited to explain a solution method for the Linear Assignment Problem. In 1969, Mack developed this method, which can be considered a forerunner of Tomizawa's (1971).

We start with a short review of assignment algorithms, followed by an exposition of elements of these algorithms. Then we explain Mack's algorithm illustrated with an example and we will show it to be near-equivalent to the Hungarian Method. Finally we indicate how Mack's method can be improved.

As is well known, the Linear Assignment Problem on the $n \times n$ cost matrix (c_{ij}) can be formulated as a linear program:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} \cdot x_{ij}$$

subject to

$$\sum_{j=1}^{n} x_{ij} = 1 \quad (i = 1,...,n),$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad (j = 1,...,n),$$

$$x_{ij} \in \{0,1\} \quad (i,j = 1,...,n).$$

The zero-one constraints on the x_{ij} can be relaxed to nonnegativity restrictions, yielding the dual problem:

$$\max \sum_{i=1}^{n} u_i + \sum_{j=1}^{n} v_j$$

subject to

$$c_{ij} - u_i - v_j \geqslant 0 \quad (i,j = 1,...,n).$$

From the dual variables u_i and v_j we may calculate the reduced costs $c_{ij} - u_i - v_j$ $(i,j = 1,...,n)$. So, the dual problem is to find a reduction of the cost matrix with maximum sum and non-negative reduced costs.

In this and the following sections, n is the number of assignments to make. Indices i and j refer to rows and columns respectively; x_i is the index of the column assigned to row i and y_j the index of the row assigned to column j, with $x_i = 0$ for an unassigned row i and $y_j = 0$ for an unassigned column j; the dual variable u_i corresponds to row i and v_j to column j. We denote the reduced costs $c_{ij} - u_i - v_j$ by $cred_{ij}$, and we may refer to the dual variables as 'prices'. An unassigned row or column is 'free'.

2. A REVIEW OF ALGORITHMS

Methods to solve the Linear Assignment Problem can be classified (roughly) in three categories:

a. algorithms based on maximum flow,
b. algorithms based on shortest paths,
c. algorithms based on the simplex method.

Most algorithms based on *maximum flow* are primal-dual methods. Papadimitriou & Steiglitz (1982) give an introduction to these methods that is very well suited for use in the classroom. A primal-dual method for the Linear Assignment Problem performs the following steps:

step 1. find a feasible dual solution;
step 2. solve a restricted primal problem, that is, find a (partial) primal solution that has complementary slackness with the dual solution;
step 3. terminate if the solution of the restricted primal problem solves the Linear Assignment Problem; otherwise, solve a restricted dual problem to adjust the dual solution leading to a less restricted primal problem, and return to step 2.

Essentially, step 2 consists of solving a maximum flow problem on an auxiliary graph. The notion 'complementary slackness' known from the theory of linear programming will be described in section 3 for the use in the context of the Linear Assignment Problem.

Historically seen Kuhn's Hungarian algorithm (1955, 1956) was actually the 'father' of the general primal-dual algorithm. The original method had computational complexity $O(n^4)$, but later $O(n^3)$ versions were developed (Lawler (1976)). Bertsekas (1981) also presented a primal-dual algorithm. The method is of the Hungarian type, and the best version even uses the Hungarian algorithm itself.

The methods based on *shortest paths* are dual algorithms in the sense that dual feasibility exists and primal feasibility has to be reached. This is achieved by considering the Linear Assignment Problem as a minimum cost flow problem, which can be solved by steps that involve finding shortest (augmenting) paths on an auxiliary graph.

In this group two algorithms, both of time complexity $O(n^3)$, stand out: Tomizawa's from 1971 and Hung & Rom's from 1980. The latter method is the more ingenious, but the former approach the faster. The algorithm of Tomizawa is initialized with a partial primal solution and a corresponding feasible dual solution. The partial assignment is augmented into a complete solution by primal steps in each of which one shortest augmenting path is determined using Dijkstra's method (1959). Hung & Rom's initial solution is complete, but may be infeasible. They determine in each step a shortest path tree, which takes more effort, but may lead to finding more augmenting paths per iteration.

Jonker & Volgenant (1987) described a Linear Assignment Problem algorithm including a Pascal implementation, that appears to be faster than the best known methods from the literature.

We highlight in this note the so-called Bradford method of Mack (1969), especially for its intuitively appealing presentation. It resembles the method of Hung & Rom, but, as originally presented, has computational complexity $O(n^4)$. Adapting it to obtain complexity $O(n^3)$ results in an algorithm close to Tomizawa's.

The *linear programming* based algorithms in the third category are (very) specialized versions of the simplex method. The best published results are from Barr, Glover & Klingman (1977). A major difficulty with all of these methods is the phenomenon of zero pivot steps. This can be illustrated in the classroom by solving a Linear Assignment Problem example as a transportation problem, showing a lot of degeneracy: almost half of the basic variables are equal to zero. A drawback is also their relatively complex implementation, as compared to the other approaches. Computational experiments (Hung & Rom (1980)) show that they are outperformed by the best primal-dual and dual algorithms.

The $O(n^3)$ Signature algorithm presented by Balinski (1985) also belongs to this category. It considers feasible dual solutions corresponding to trees in the bipartite graph of row and column nodes. Since its first publication, some refinements have been published, but, up to now, no computational results have been presented.

3. Elements of linear assignment algorithms

Most primal-dual and dual Linear Assignment Problem algorithms are based on only a few standard operations:

- *initialization:* a feasible dual solution u_i ($i = 1,...,n$) and v_j ($j = 1,...,n$) is determined; the primal solution x_i ($i = 1,...,n$), with corresponding y_j ($j = 1,...,n$), is initialized so that complementary slackness holds, that is, $x_i = j$ only if $cred_{ij} = 0$;
- *finding an augmenting path:* a sequence of, alternately, row and column indices is determined, with the first an unassigned row, the last an unassigned column, and the intermediate columns and rows assigned in successive pairs;
- *augmentation:* augmentation of a partial solution can take place along an augmenting path by assigning all rows in the path to their succeeding column, which results in one more assignment;
- *adjustment of the dual solution:* prices are adjusted either to obtain at least one additional zero reduced cost coefficient, while maintaining complementary slackness, or to restore complementary slackness after augmentation of a partial assignment.

The concept of augmentation along alternating paths is the basis for every assignment algorithm. How to adjust the dual solution merits some thought. Each algorithm specifies its own rules for this operation. The purpose is to maintain complementary slackness, that is,

$$c_{ij} - u_i - v_j \geq 0 \qquad (i,j = 1,...,n), \tag{1}$$

$$c_{ik} - u_i - v_k = 0, \text{ if } x_i = k \quad (i = 1,...,n). \tag{2}$$

Substituting the u_i from (2) into (1) leads to

$$c_{ik} - v_k \leq c_{ij} - v_j \quad (j = 1,...,n).$$

This implies that for every assigned column k ($y_k = i$) the v_k must be chosen so that

$$c_{ik} - v_k = \min\{c_i - v_j \mid j = 1,...,n\},$$

and for every assigned row i ($x_i = k$) the u_i must be set at

$$u_i = c_{ik} - v_k.$$

So all assignments in a (partial) solution must correspond to row minima in the reduced costs matrix. After augmentation of a partial solution, this trivial observation will usually show the best way to adjust prices. It even forms the basis of the assignment algorithm of Mack (1969) in which algorithm only the values of the v-variables have to be recorded, making it easier to understand the solution procedure.

4. The algorithm of Mack

Mack's linear assignment method is easy to understand, and easy to use. We will show it to be near-equivalent to both the algorithm of Tomizawa and the Hungarian method. So, in a way, Mack's algorithm provides the best statement for an Hungarian-type assignment algorithm. The method is based on two trivial observations for the Linear Assignment Problem:

1. the cost matrix can be reduced without influencing the optimal solution;
2. an optimal solution is found if for a certain reduced cost matrix the row minima occur in different columns.

As long as 2 is not fulfilled, Mack's algorithm adjusts the reduced costs in such a way that the row minima are spread over more columns.

3*	7	6	6	6*	7	6	6	6	7	6*	6
1*	6	8	8	4*	6	8	8	4*	6	8	8
3	0*	8	1	6	0*	8	1	6	0*	8	1
0*	7	9	9	3*	7	9	9	3*	7	9	9
	(a) start				(b)				(c)		

8	7	6*	6	9	8	6*	6	9	8	6*	6
6*	6	8	8	7*	7	8	8	7	7*	8	8
8	0*	8	1	9	1*	8	1	9	1	8	1*
5*	7	9	9	6*	8	9	9	6*	8	9	9
	(d)				(e)				(f) optimal		

Figure 1. An assignment problem solved by Mack's algorithm (bases are starred).

We illustrate the method on a simple example, which has the cost matrix given in Figure 1(a). The column where for a certain row i the minimum current reduced costs occur is called the *base* of row i, denoted base_i. (The bases are starred in Figure 1.) The method terminates if every column contains one base.

First consider column 1, which contains more than one base. By increasing its entries we create an alternative position for one of the bases in this column. If column 1 is increased by 3, an alternative position is found for base_1, underlined in matrix (b). This position being free, we switch base_1 from column 1 to column 3 in matrix (c), using in fact an augmenting path of length two. In the next iteration we consider again column 1, still containing two bases. After increasing column 1 by the amount 2, we find a new possibility for base_2 in matrix (d). The corresponding column 2 is already occupied by a base, so from now on we must increase columns 1 and 2 simultaneously. An increase of 1 for both columns leads to the cost matrix of Figure 1(e). In this matrix a path exists, alternating between bases and alternative bases, along which the current set of bases can be spread over one more column. The solution in Figure 1(f) is optimal. Note that after every step, in (c) and in (f), the bases are spread over one more column.

This is a more formal statement of the *assignment algorithm of Mack.*

step 1. *Initialization:* Determine the bases in the cost matrix.

step 2. *Termination,* if every column contains one base.

step 3. Select a column j that contains more than one base.
Set COL := $\{j\}$ and ROW := $\{i \mid \text{base}_i \in \text{COL}\}$.

step 4. For $i \in$ ROW set $m_i = \min\{cred_{ik} - cred_{i,\text{base}_i} \mid k \in \text{COL}\}$.
Determine $\delta = \min\{m_i \mid i \in \text{ROW}\}$.
Let rr be a row and kk a column for which δ is assumed.

step 5. *Adjust the dual solution:* Increase the reduced costs of all columns in COL by the amount δ.
If column kk contains no base, go to step 6. Otherwise go to step 7.

step 6. *Augmentation:* A path has been found consisting of, alternately, bases and alternative bases, starting in column kk:
- alter the current set of bases along the alternating path,
- go to step 2.

step 7. Column kk is base for some row(s):
- mark column kk as alternative base for row rr,
- COL := COL $\cup \{kk\}$; ROW := ROW $\cup \{i \mid \text{base}_i = kk\}$,
- go to step 4.

Bunday and Garside (1987) have published a computer program for Mack's method.

5. Improving Mack's algorithm

When considering the complexity of Mack's algorithm, step 4 turns out to be inefficient. Its formulation above requires $O(n^2)$ operations to update the row minima over the columns not in COL. An alternative formulation is:

step 4. For $k \in$ COL set $mm_k = \min\{cred_{ik} - cred_{i,\text{base}_i} \mid i \in \text{ROW}\}$.
Determine $\delta = \min\{mm_k \mid k \in \text{COL}\}$.
Let rr be a row and kk a column for which δ is assumed.

When δsum is the sum of the δ-values by which columns in COL were increased in previous applications of step 5 during the current iteration, we note that

$$mm_k + \delta\text{sum} = d_k \quad (k = 1,...,n).$$

This implies that updating the row minima is equivalent to determining a shortest path from the original column j (from step 2) to any unoccupied column in an auxiliary network. As in Tomizawa, Dijkstra's method can be used as we consider the (non-negative) reduced costs. So in this formulation, the column minima over the rows in ROW can be updated in $O(n)$ operations, and the computational complexity of the method becomes $O(n^3)$.

Using the second formulation of step 4 yields a method that is almost equivalent to that of Tomizawa (1971). Note, however, that Mack's method was published in 1969. Tomizawa's method is obtained as follows:

- rows are assigned only if the minimum reduced costs occur in an unassigned

column, that is, every column is assigned at most once;
- the set ROW is initialized (in step 3) with any unassigned row and COL with the empty set.

The improvement of this section may be omitted when treating Mack's method in the classroom. For further improvements, e.g., about finding an initial solution (step 1), we refer to Jonker & Volgenant (1987).

6. Conclusion

We have highlighted Mack's Linear Assignment Problem algorithm to promote it as a method to be discussed in textbooks on Operations Research and Mathematical Programming. We think the given explanation as well as the cohesion with other Linear Assignment Problem algorithms to contain enough arguments for the recommendation to use the method for teaching students about the Linear Assignment Problem.

References

M.L. Balinski (1985), *Signature methods for the assignment problem*, Operations Research 33, 527-536.

R. Barr, F. Glover and D. Klingman (1977), *The alternating path basis algorithm for assignment problems*, Mathematical Programming 13, 1-13.

D.P. Bertsekas (1981), *A new algorithm for the linear assignment problem*, Mathematical Programming 21, 152-171.

B.D. Bunday, G.R. Garside (1987), *Linear programming in Pascal*, Edward Arnold, London.

H.G. Daellenbach, J.A. George and D.C. McNickle (1983), *Introduction to Operations Research Techniques*, second edition, Allyn and Bacon, Boston.

E.W. Dijkstra (1959), *A note on two problems in connexion with graphs*, Numerische Mathematik 1, 269-271.

M.S. Hung and W.O. Rom (1980), *Solving the assignment problem by relaxation*, Operations Research 28, 969-982.

R. Jonker and A. Volgenant (1987), *A shortest augmenting path algorithm for dense and sparse linear assignment problems*, Computing 38, 325-340.

H.W. Kuhn (1955), *The Hungarian method for the assignment problem*, Naval Research Logistics Quarterly 2, 83-97.

H.W. Kuhn (1956), *Variants of the Hungarian method for assignment problems*, Naval Research Logistics Quarterly 2, 253-258.

E.L. Lawler (1976), *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart & Winston, New York.

C. Mack (1969), *The Bradford method for the assignment problem*, New Journal of Statistics and Operational Research 1, 17-29.

C.H. Papadimitriou and K. Steiglitz (1982), *Combinatorial Optimization: Algorithms and Complexity*, Prentice Hall, Englewood Cliffs, New Jersey.

H.A. Taha (1987), *Operations Research: An Introduction*, fourth edition, Macmillan, New York.

N. Tomizawa (1971), *On some techniques useful for the solution of transportation problems*, Networks 1, 173-194.