An Extension of the Munkres Algorithm for the Assignment Problem to Rectangular Matrices

François Bourgeois and Jean-Claude Lassalle
CERN, Geneva, Switzerland

The assignment problem, together with Munkres proposed algorithm for its solution in square matrices, is presented first. Then the authors develop an extension of this algorithm which permits a solution for rectangular matrices.

Timing results obtained by using an adapted version of Silver's Algol procedure are discussed, and a relation between solution time and problem size is given.

Key Words and Phrases: operations research, optimization theory, assignment problem, rectangular matrices, algorithm

CR Categories: 5.39, 5.40

The procedure described here was developed for use in spark chamber event recognition problems occurring in high-energy nuclear physics [1]. This paper gives the theoretical background of Algorithm 415, Algorithm for the Assignment Problems (Rectangular Matrices), by the same authors, appearing on pages 805–806 of this issue.

1. The Assignment Problem (Square Matrices)

1.1 Mathematical Statement

Given the $n \times n$ matrix (a_{ij}) of real numbers, find a permutation p $(p_i; i = 1, n)$ of the integers $1, 2, \ldots, n$ that minimizes $\sum_{i=1}^{n} a_{ip_i}$.

Example. For the 3×3 matrix

$$(a_{ij}) = \begin{vmatrix} 7 & 5 & 11.2 \\ 5 & 4 & 1 \\ 9.3 & 3 & 2 \end{vmatrix}$$

there are six possible permutations for which the associated sums are

	p	$\sum_{i=1}^{n} a_{ip_i}$
(1)	1 2 3	13.0
(2)	1 3 2	11.0
(3)	2 1 3	12.0
(4)	2 3 1	15.3
(5)	3 1 2	19.2
(6)	3 2 1	24.5

Permutation (2) gives the smallest sum ($a_{11} + a_{23} + a_{32} = 11.0$), and is the solution to the assignment problem for this matrix.

1.2 An Algorithm for the Assignment Problem (J. Munkres, 1957)

To give the reader an understanding of the improvements proposed below (Section 2), we summarize the articles by Munkres [2] and Silver [3].

Most algorithms for the assignment problem, especially the one described in this section, are based on Theorems 1 and 2.

THEOREM 1. *Given a column vector (c_i) and a row vector (r_j), the square matrix (b_{ij}) with the elements $b_{ij} = a_{ij} - c_i - r_j$ has the same optimal assignment solution as the matrix (a_{ij}).*

Such (a_{ij}) and (b_{ij}) are said to be *equivalent*.

PROOF. Let p be a permutation of the integers $1, 2, \ldots, n$ minimizing $\sum_{i=1}^{n} a_{ip_i}$, then

$$\sum_{i=1}^{n} b_{ip_i} = \sum_{i=1}^{n} a_{ip_i} - \sum_{i=1}^{n} c_i - \sum_{i=1}^{n} r_{p_i}.$$

Values of the last two terms are independent of p so that if p minimizes $\sum_{i=1}^{n} a_{ip_i}$, it also minimizes $\sum_{i=1}^{n} b_{ip_i}$.

Definition. A set of elements of a matrix are *independent* if none of them occupies the same row or column.

Thus, a_{13} and a_{21} are independent elements while a_{13} and a_{23} are not. The assignment problem can thus be stated as finding a set of n independent elements of the given $n \times n$ matrix (a_{ij}) so that the sum of these elements is minimum.

THEOREM 2 (König). *Given a matrix, (a_{ij}), if m is the maximum number of independent zero elements in this matrix, then there are m lines (row or columns or both) which contain all the zero elements of (a_{ij}).*

(A proof of this theorem is presented by Berge [4].)

A statement of Munkres' algorithm follows. Briefly, it consists in finding an $n \times n$ matrix (b_{ij}) equivalent to the initial one (a_{ij}) with all $b_{ij} \geq 0$ having n independent zero elements. In the statement zero elements are distinguished by asterisks and primes—the "starred zeros" form an independent set of zeros, while the "primed zeros" are possible candidates for this set.

Some lines are distinguished too: they are said to be *covered*. A zero is called covered (noncovered) if it is in a covered (noncovered) line.

Preliminaries. (a) No lines are covered; no zeros are starred or primed. (b) For each row of the matrix (a_{ij}) subtract the value of the smallest element from each element in the row. (c) For each column of the resulting matrix subtract the value of the smallest element from each element.

Step 1. Find a zero, Z, of the matrix. If there is no starred zero in its row nor its column, star Z. Repeat for each zero of the matrix. Go to step 2.

Step 2. Cover every column containing a 0*. If all columns are covered, the starred zeros form the desired independent set; Exit. Otherwise, go to step 3.

Step 3. Choose a noncovered zero and prime it; then consider the row containing it. If there is no starred zero Z in this row, go to step 4. If there is a starred zero Z in this row, cover this row and uncover the column of Z. Repeat until all zeros are covered. Go to step 5.

Step 4. There is a sequence of alternating starred and primed zeros constructed as follows: let Z_0 denote the uncovered 0′. Let Z_1 denote the 0* in Z_0's column (if any). Let Z_2 denote the 0′ in Z_1's row. Continue in a similar way until the sequence stops at a 0′, Z_{2k}, which has no 0* in its column. Unstar each starred zero of the sequence, and star each primed zero of the sequence. Erase all primes and uncover every line. Return to step 2.

Step 5. Let h denote the smallest noncovered element of the matrix; it will be positive. Add h to each covered row; then subtract h from each uncovered column. Return to step 3 without altering any asterisks, primes, or covered lines.

Preliminaries and step 5 find a matrix equivalent to the previous one. Theorem 1 is applied in the preliminaries by: (a) constructing the column vector (c_i) by the rule: c_i = smallest element in row i of (a_{ij}); and (b) constructing the row vector (r_j) by the rule: r_j = smallest element in column j of the new matrix (a_{ij}), $(a_{ij} = a_{ij} - c_i)$. In step 5 the (c_i) and (r_j) are constructed by the rules: $c_i = h$ or 0 depending on whether row i is covered or not; $r_j = 0$ or $-h$ depending on whether column j is covered or not.

Steps 2 to 4 are a constructive procedure to find, in the current matrix, a maximal set of independent zeros and a minimal set of lines which contain all zeros that Theorem 2 states must exist.

The maximum number of operations required by this procedure is proportional to n^3 and is smaller than the $n!$ operations needed to find all possible permutations.

Example. For the 3×3 matrix

$$(a_{ij}) = \begin{vmatrix} 7 & 5 & 11.2 \\ 5 & 4 & 1 \\ 9.3 & 3 & 2 \end{vmatrix}$$

The sequence of equivalent matrices is:

$$\text{preliminaries} \begin{vmatrix} 2 & 0 & 6.2 \\ 4 & 3 & 0 \\ 7.3 & 1 & 0 \end{vmatrix} \quad \begin{vmatrix} 0 & 0 & 6.2 \\ 2 & 3 & 0 \\ 5.3 & 1 & 0 \end{vmatrix}$$

$$\text{step 5} \begin{vmatrix} 0^* & 0' & 6.2 \\ 2 & 3 & 0^* \\ 5.3 & 1 & 0 \end{vmatrix} \rightarrow \begin{vmatrix} 0^* & 0' & 6.2 \\ 1 & 2 & 0^* \\ 4.3 & 0 & 0 \end{vmatrix}$$

$$\text{step 2 (end)} \begin{vmatrix} 0^* & 0 & 6.2 \\ 1 & 2 & 0^* \\ 4.3 & 0^* & 0 \end{vmatrix}$$

2. Extension of the Assignment Problem to Rectangular Matrices

2.1 Mathematical Statement

The assignment problem can be extended to rectangular matrices and may then be formulated as follows.

Given the $n \times m$ matrix (a_{ij}) of real numbers, find a set of k independent elements $[k = \min(n, m)]$ so that the sum of these elements is minimum.

Example. For the 2×3 matrix

$$(a_{ij}) = \begin{vmatrix} 7 & 5 & 11.2 \\ 5 & 4 & 1 \end{vmatrix}$$

there are six possible sets of two independent elements—$(a_{11}\ a_{22})$, $(a_{11}\ a_{23})$, $(a_{12}\ a_{21})$, $(a_{12}\ a_{23})$, $(a_{13}\ a_{21})$ and $a_{13}\ a_{22})$. The smallest sum is $a_{12} + a_{23} = 6.0$.

2.2 Algorithm

Consider

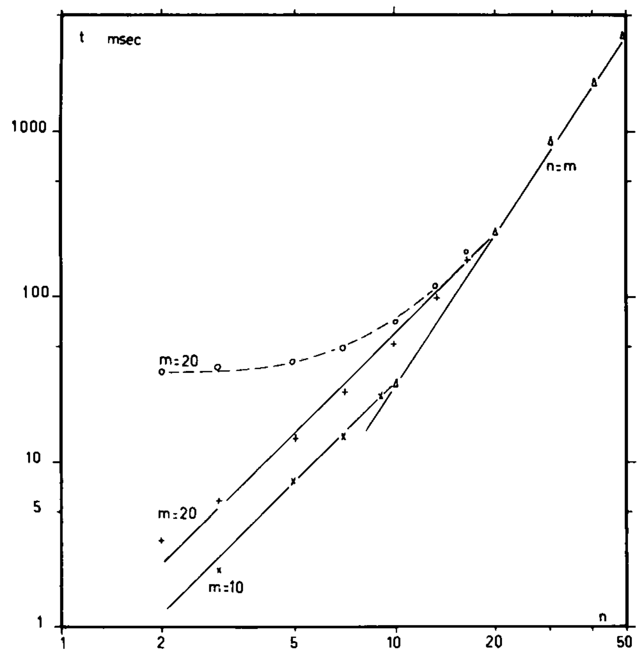
$$k = \min(n, m)$$
$$l = \max(n, m).$$

Then to each set of k independent elements corresponds a permutation of the integers $1, 2, \ldots, l$ taken k at a time. As there are $l!/(l-k)!$ such permutations it is, for large values of l and k, expensive to calculate all the different sums.

Until now, the accepted way of solving this problem [5] has been to add $l - k$ lines of zero elements to the

Fig. 1. Time taken by assignment algorithms to find an optimal solution as a function of n ($n \times m$ random matrices. Computer: CDC 6500)

$\triangle$: computation times observed with $n = m$;
$\times$: computation times observed with $m = 10$;
$+$: computation time observed with $m = 20$;
$\bigcirc$: computation times observed with $m = 20$, matrix squared with zeros;
Solid line : $0.03 \times k^2 \times l$ time variation.



given matrix and apply the Munkres algorithm. This is both space and time consuming (especially when k and l are very different).

A better way of adapting the Munkres algorithm has been derived as follows. (First, two remarks: (i) In the algorithm, preliminaries may be skipped or partially executed without altering the final result; (ii) Whenever step 4 is executed, the number, n_s, of starred zeros is increased by one ($n_{s+1} = n_s + 1$), but the set of lines containing n_s of them remains the same; i.e. only n_s of these lines are covered.)

Let (b_{ij}) be a $l \times l$ matrix obtained by adding $l - k$ lines of the same element α to the initial $n \times m$ matrix (a_{ij}). The matrix (b_{ij}) may be divided into two submatrices: $A \equiv (a_{ij})$ and C such that $c_{ij} = \alpha$.

$$B = (AC) \quad \text{when } n > m$$

$$B = \begin{pmatrix} A \\ C \end{pmatrix} \quad \text{when } n < m.$$

It is readily seen that α can be chosen large enough so that no zero will appear in submatrix C before k zeros are starred in the "A" part.

Moreover, at least $(l - k)^2$ elements of submatrix C have never been covered (see remark (ii), above) and these are the smallest of the noncovered part of B.

Then, steps 1 to 4 are executed and $l - k$ starred zeros can be added without altering the previous set of k independent zeros. Thus, it is not necessary to generate the square matrix (b_{ij}).

The statement of the extended algorithm follows.

Preliminaries. (a) $k = \min(n, m)$, no lines are covered, no zeros are starred or primed. (b) *If the number of rows is greater than the number of columns, go at once to step* 0. Consider a particular row of the matrix (a_{ij}); subtract the smallest element from each element in the row; do the same for all other rows. *If the number of columns is greater than the number of rows, go at once to step* 1.

Step 0. (c) For each column of the resulting matrix, subtract from each entry the smallest entry in the column.

Step 1. Remains the same.

Step 2. Cover every column containing a 0*. If k columns are covered, the starred zeros form the desired independent set. Otherwise, go to step 3.

Steps 3 to 5. Remain the same.

3. Conclusion

R. Silver's program implementing Munkres algorithm for the assignment problem has been modified for rectangular matrices. Whatever the matrix A may be, our procedure is faster and requires less memory space than the one which consists in adding $l - k$ lines of zero elements [5].

To a good approximation, the mean time per matrix is: $t = \beta \times k^2 \times l$, where $k = \min(n, m)$, $l = \max(n, m)$, and β = constant of proportionality. For very nonsquare matrices, i.e. $l > 2 \times k$, the procedure can give an improvement factor as large as 10.

The FORTRAN version of the program running on the CERN CDC 6500 requires one thousand 60-bit words plus the space required for the assignment matrix (maximum size 100×100) and takes about $t \simeq 0.03 \times k^2 \times l$ milliseconds of central processor time.

Calculations times have been plotted in Figure 1 for the 3 cases of $m = n$, $m = 10$, and $m = 20$ with n varying from 2 to 50. To illustrate the increased efficiency, times have also been plotted for the zero-completed procedure with $m = 20$.

Received September 1970; revised November 1970 and January 1971

References

1. Bourgeois, F., Grote, H., and Lassalle, J.C. Pattern recognition methods for Omega and SFM Spark Chamber experiments. CERN-Data Handling Division DD/DH/70/13, Geneva, Switzerland, Mar. 1970.
2. Munkres, J. Algorithms for the assignment and transportation Problems. *J. Siam 5* (Mar. 1957), 32–38.
3. Silver, R. An algorithm for the assignment problem. *Comm. ACM 3*, (Nov. 1960), 605–606.
4. Berge, C. *Théorie des Graphes et ses Applications.* Dunod, Paris, 1958.
5. Kaufmann, A. *Introduction à la Combinatorique en Vue de ses Applications.* Dunod, Paris, 1968, pp. 506–515.