AN ANALYSIS OF A DECOMPOSITION HEURISTIC FOR THE ASSIGNMENT PROBLEM

David AVIS and Luc DEVROYE

School of Computer Science, McGill University, Montreal, Quebec, Canada

Received May 1984
Revised November 1984

J.M. Kurtzberg proposed a method of obtaining approximate solutions to the assignment problem by decomposing a large problem into many smaller subproblems. Thus a $km \times km$ assignment problem is decomposed into k^2 problems of size $m \times m$ and one problem of size $k \times k$. In this paper we analyze the performance of this heuristic, obtaining the following main results:

(1) For the maximization problem, the ratio of the optimal solution to the heuristic solution can be as large as, but cannot exceed $\min(k, m)$;

(2) For the minimization problem, if $k = o(n/\log n)$ where $n = mk$, and the matrix elements are independently drawn from a uniform distribution on (0, 1), in the limit the expected value of the heuristic solution is at least $k/3$ times that of the optimal solution.

assignment problem • heuristics • analysis of algorithms

OR/MS Index: 486, 632.

1. Introduction

Let $A = (a_{ij})$ be an $n \times n$ matrix whose entries are nonnegative real numbers. The maximum (resp. minimum) weight assignment problem is to choose precisely one element from each row and column so that total sum is maximized (resp. minimized). Many techniques are known for finding an optimal solution in time $O(n^3)$ and space $O(n^2)$. See for example [15].

In this case where n is large, time and/or space limitations may make it infeasible to find an optimal solution, hence the need for heuristics. J.P. Kurtzberg [3] has proposed a number of heuristics for the assignment problem. One of the most interesting involves partitioning the matrix into smaller submatrices and then solving many smaller assignment problems. These solutions are then used to solve one 'master' problem that provides an approximate solution to the original problem. This type of approach has been found useful for a variety of large scale optimization problems. In this paper we provide an analysis of the quality of the approximate solutions produced by this heuristic. For a comparison with other heuristics for the assignment problem, the reader is referred to the survey paper [1]. We begin by stating the heuristic more precisely and discussing its complexity.

Research supported by N.S.E.R.C. grants A3013 and A3456.

For convenience, we assume that $n = k \times m$ for integers k and m. This can always be achieved by enlarging A if necessary and by setting the new matrix elements to zero in case of maximization or some very large positive value in case of minimization. The solution to an assignment problem on A can be described by an $n \times n$ *permutation matrix* $M = (m_{ij})$. The entries in M are either zero or one and each row and column of M contains precisely one nonzero entry, corresponding to the element chosen in the assignment. The *value* of the solution is then the matrix product $A \times M$.

procedure ASSIGN (A, k, m)
  begin
  1. Partition A into k^2 submatrices R_{ij}, $1 \leqslant i, j \leqslant k$ of size $m \times m$;
  2. Solve the assignment problem on each R_{ij}

using an optimal algorithm obtaining solutions with value d_{ij} and permutation matrices Y_{ij};
3. Solve the $k \times k$ assignment problem $D = (d_{ij})$ obtaining permutation matrix $Q = (q_{ij})$;
4. Build the $n \times n$ permutation matrix M from k^2 submatrices $q_{ij}Y_{ij}$, $1 \leqslant i, j, \leqslant k$.

end.

The algorithm is illustrated schematically in Figure 1. The complexity analysis of ASSIGN is straight forward. Step 2 requires the solution of k^2 assignment problems of size $m \times m$ and hence requires $O(k^2m^3)$ time. Step 3 requires the solution of one $k \times k$ assignment problem and hence requires $O(k^3)$ time. A nice feature of the algorithm is that A may be stored in secondary storage and the main memory required is only $\max(m^2, k^2)$. Therefore the *space-optimal* version of ASSIGN is obtained when $m = k = \sqrt{n}$ and requires $O(n)$ space and $O(n^{2.5})$ time. The *time-optimal* version of ASSIGN is obtained by setting $k^2m^3 = k^3$ and hence $k = n^{3/4}$. This version has space requirement $O(n^{1.5})$ and time requirement $O(n^{2.25})$.

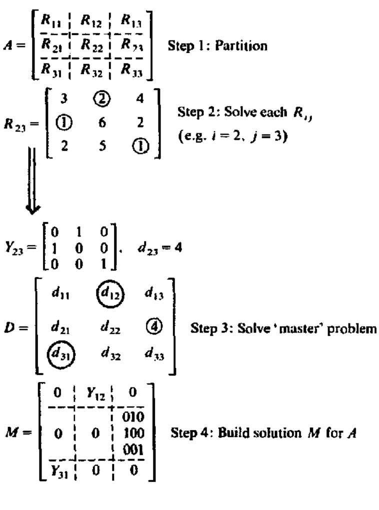

Fig. 1. Schematic illustration of ASSIGN: Minimization ($k = 3$, $m = 3$).

In the next two sections we analyze the quality of the solution obtained by ASSIGN. The next section is concerned with worst case analysis, and the last section with expected case analysis.

2. Worst case analysis

In this section, we consider the worst case analysis of the heuristic ASSIGN. We denote by $H^{m,k}_{\max}(A)$ and $H^{m,k}_{\min}(A)$ the value obtained by ASSIGN(A, k, m) for the maximization and minimization problems respectively. Let $O_{\max}(A)$ and $O_{\min}(A)$ be the respective optimal solutions. One measure of the effectiveness of the heuristic are the *ratio bounds*

$$\frac{O_{\max}(A)}{H^{m,k}_{\max}(A)} \quad \text{and} \quad \frac{H^{m,k}_{\min}(A)}{O_{\min}(A)}.$$

In this section we analyze how large these ratios can get in the worst case. Unless stated otherwise, all matrix subscripts are reduced mod n to the range $1, 2, \ldots, n$ in this section.

We begin by defining a matrix that will be useful in the worst case analysis. Again, for convenience, we assume $n = km$ for integers k, m. Consider the matrix $W_{m,k} = (w_{ij})$, $1 \leqslant i, j \leqslant n$ defined by

$$w_{ij} = \begin{cases} 1 & i = am + b + 1,\ j = bm + i, \\ & a = 0, 1, \ldots, k-1,\ b = 0, 1, \ldots, m-1, \\ 0 & \text{otherwise.} \end{cases}$$

For $n = 6$, The constructions for $m = 2$, $k = 3$ and $m = 3$, $k = 2$ are given in Figure 2. Interchanging the values zero and one in $W_{m,k}$ yields the complementary matrix denoted $\overline{W}_{m,k}$. The required properties of these matrices are contained in the following lemma:

$$\left[\begin{array}{cc|cc|cc} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{array}\right] \qquad \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right]$$

$W_{2,3}$ $\qquad$ $W_{3,2}$

Fig. 2.

Lemma 1. (a) $O_{\max}(W_{m,k}) = km$. (b) $O_{\min}(\overline{W}_{m,k}) = 0$. (c) $H_{\max}^{m,k}(W_{m,k}) = k$ if $k \geqslant m$. (d) $H_{\min}^{m,k}(\overline{W}_{m,k}) \geqslant k$ if m, $k \geqslant 2$.

Proof. We begin by showing that $W_{m,k}$ is a permutation matrix. First observe that i takes the values $1, 2, \ldots, n$ as we step through values of a letting b run from 0 to $m-1$. Thus it suffices to show that the same value of j cannot occur twice. Suppose, on the contrary, that

$$b_1 m + (a_1 m + b_1 + 1) = b_2 m + (a_2 m + b_2 + 1)$$

for integers $0 \leqslant a_1, a_2 \leqslant k-1$ and $0 \leqslant b_1, b_2 \leqslant m-1$. This implies that

$$(m+1)(b_1 - b_2) = (a_2 - a_1)m \pmod{mk},$$

and hence that $b_1 - b_2$ is a multiple of m. This is only possible if $b_1 = b_2$. Therefore

$$(a_2 - a_1)m = 0 \pmod{mk},$$

and so $a_2 - a_1$ is a multiple of k. This implies that $a_1 = a_2$, completing the proof that $W_{m,k}$ is a permutation matrix. This proves parts (a) and (b) of the lemma.

Consider the partition of $W_{m,k}$ into k^2 submatrices R_{st} of size $m \times m$ in step 1 of ASSIGN. We will determine the number of nonzero entries in each submatrix. Set $a = 0$ and let b range from 0 to $m-1$. These m nonzero elements are distributed into $R_{11}, R_{12}, \ldots, R_{1k}$ as follows:

$$R_{11}[1, 1] = 1, \quad R_{12}[2, 2] = 1, \quad R_{13}[3, 3] = 1, \ldots.$$

If $k \geqslant m$, this sequence ends with the entry $R_{1m}[m, m]$. Otherwise it continues:

$$R_{11}[k+1, k+1] = 1, \quad R_{12}[k+2, k+2] = 1, \ldots$$

terminating with $R_{1, m \bmod k}[m, m] = 1$. Thus each submatrix receives either $\lfloor \frac{m}{k} \rfloor$ or $\lceil \frac{m}{k} \rceil$ nonzero entries. The same analysis and results holds for each value of a, and hence for each submatrix R_{st}.

If $k \geqslant m$, then each submatrix receives at most one nonzero entry, hence part (c) of the lemma follows. If m, $k \geqslant 2$, then no submatrix R_{st} contains as many as m nonzero elements. Therefore in the complementary matrix $\overline{R}_{st}$, there are less than m zeroes and the minimum assignment in this submatrix has weight at least one. Thus (d) follows and the proof of the lemma is complete. □

When $m \geqslant k$ we define the matrix $W_{m,k}^*$ as follows:

$$w_{ij}^* = \begin{cases} 1 & i = am + b + 1, \; j = bm + i, \\ & a = 0, \ldots, k-1, \; b = 0, \ldots, k-1, \\ 0 & \text{otherwise.} \end{cases}$$

W_{mk}^* is a zero-one matrix whose nonzero entries are a subset of those of $W_{m,k}$. Analogously to Lemma 1, we can prove the following lemma.

Lemma 2. (a) $O_{\max}(W_{m,k}^*) = k^2$, (b) $H_{\max}^{m,k}(W_{m,k}^*) = k$.

Let $\mathscr{A}_n$ be the set of $n \times n$ matrices which are nonzero and whose entries are nonnegative real numbers. Then we can prove the main theorem of this section.

Theorem 1

(a) $$\sup_{A \in \mathscr{A}_n} \frac{O_{\max}(A)}{H_{\max}^{m,k}(A)} = \min(k, m),$$

(b) $$\sup_{A \in \mathscr{A}_n} \frac{H_{\min}^{m,k}(A)}{O_{\min}(A)} = +\infty.$$

Proof. Part (b) follows immediately from Lemma 1 parts (b) and (d). From Lemma 1 parts (a) and (c) and from Lemma 2 we can conclude that

$$\sup_{A \in \mathscr{A}_n} \frac{O_{\max}(A)}{H_{\max}^{m,k}(A)} \geqslant \min(k, m).$$

Therefore, to establish part (a) of the theorem, it suffices to prove the above with the inequality reversed. Fix $n = mk$ and let A be any matrix in $\mathscr{A}_n$. Referring to the heuristic ASSIGN, we see that

$$O_{\max}(A) \leqslant \sum_{i=1}^{k} \sum_{j=1}^{k} d_{ij},$$

since d_{ij} must be larger than the sum of all elements in the optimal assignment for A that lie in the submatrix R_{ij}. Now every $k \times k$ matrix D contains k disjoint assignments that partition the elements of D. Therefore, by averaging,

$$H_{\max}^{m,k}(A) \geqslant \frac{\sum_{i=1}^{k} \sum_{j=1}^{k} d_{ij}}{k} \geqslant \frac{O_{\max}(A)}{k}$$

which shows that the worst case ratio is bounded by k. Now consider the matrix $D^* = (d_{ij}^*)$, $1 \leqslant i, j \leqslant k$, defined by

$$d_{ij}^* = \begin{cases} d_{ij} & \text{if } R_{ij} \text{ contains an element in the optimal assignment} \\ 0 & \text{otherwise.} \end{cases}$$

Then D^* can have at most m positive elements in each row and column. These positive entries can be decomposed into at most m disjoint assignments. Since

$$\sum_{i=1}^{k} \sum_{j=1}^{k} d_{ij}^* \geqslant O_{\max}(A)$$

we have

$$H_{\max}^{m,k}(A) \geqslant \frac{\Sigma_{i,j} d_{ij}^*}{m} \geqslant \frac{O_{\max}(A)}{m},$$

which proves that the worst case ratio is bounded by m. □

In order to put this result in perspective, consider the heuristic GREEDY which repeatedly chooses the largest (resp. smallest) matrix element that does not lie in the same row or column as any previously chosen element. Then, as shown in [1], the maximization problem has a ratio bound of 2, and the minimization problem has an unbounded ratio. GREEDY runs in time $O(n^2 \log n)$.

3. Expected case analysis

In this section we give a probabilistic analysis of ASSIGN. Throughout this section, we assume that A is an $n \times n$ matrix whose elements are independent uniformly distributed random variables on the interval [0, 1]. Under these assumptions, the minimization and maximization problems are symmetric, so that

$$E(O_{\max}(A)) = n - E(O_{\min}(A)),$$
$$E(H_{\max}^{m,k}(A)) = n - E(H_{\min}^{m,k}(A)).$$

We therefore consider just the minimization problem and denote the random variables corresponding to the value of the optimal and heuristic solutions by O^n and $H^{m,k}$ respectively. The expected value of O^n has been obtained by Walkup, and we state the rather surprising result here.

Theorem 2 [6]. $E(O^n) \leqslant 3$.

An application of Theorem 2 yields an upper bound on ASSIGN:

$$E(H^{m,k}) \leqslant 3k.$$

We will show in the next theorem that for most values of m and k, this bound is sharp to a constant factor.

Theorem 3. *If* $k = e^{o(m)}$ *and* $m \to \infty$ *(which is satisfied if* $k = o(n/\log n)$*), then*

$$\liminf_{m,k \to \infty} \frac{E(H^{m,k})}{k} \geqslant 1.$$

Proof. Let $R_{ij}[r, s]$, $1 \leqslant i, j \leqslant k$, $1 \leqslant r, s \leqslant m$ be the element $[r, s]$ of submatrix R_{ij}. A simple argument shows that

$$H^{m,k} \geqslant \sum_{i=1}^{k} \min_{1 \leqslant j \leqslant k} \sum_{r=1}^{m} \min_{1 \leqslant s \leqslant m} R_{ij}[r, s].$$

We claim that the random variable $\min_{1 \leqslant s \leqslant m} R_{ij}[r, s]$ is stochastically greater than

$$E_{ij}(r)/(E_{ij}(r) + m),$$

where $E_{ij}(r)$ is an independent exponential random variable. Indeed, dropping the subscripts i, j and the index r, we have, for x contained in [0, 1],

$$P\Big(\min_{1 \leqslant s \leqslant m} R_{ij}[r, s] > x\Big) = (1-x)^m$$
$$\geqslant e^{-mx/(1-x)} = P\left(\frac{E}{E+m} > x\right).$$

Thus, for all constants c in the open interval (0, 1),

$$E(H^{m,k}) \geqslant kE\left(\min_{1 \leqslant j \leqslant k} \sum_{r=1}^{m} \frac{E_{1j}(r)}{E_{1j}(r) + m}\right)$$
$$\geqslant kcP\left(\sum_{r=1}^{m} \frac{E(r)}{E(r)+m} \geqslant c\right)^k = kc\beta.$$

If we can prove that $\lim_{n \to \infty} \beta = 1$, then, since c was arbitrary, the proof is complete. We have

$$\beta = P\left(\sum_{r=1}^{m} \frac{E(r)}{E(r)+m} \geqslant c\right)^k$$
$$\geqslant 1 - kP\left(\sum_{r=1}^{m} \frac{E(r)}{E(r)+m} < c\right)$$
$$\geqslant 1 - kP\left(\sum_{r=1}^{m} E(r) I_{[E(r) < m]} < 2cm\right)$$
$$\geqslant 1 - k\left[P\left(\sum_{r=1}^{m} E(r) < cm\right) + P\left(\sum_{r=1}^{m} E(r) I_{[E(r) \geqslant m]} > cm\right)\right].$$

Applying Chebyshev's inequality to the last term, we have

$$\beta \geqslant 1 - k\left[P\left(\frac{G_n - m}{m} < c - 1\right) + \frac{1}{c}\int_m x\, e^{-x} dx\right],$$

where G_m is a gamma distributed random variable with parameter m. Finally, using a strong tail inequality for gamma distributions (see, for example Devroye [2]), we obtain

$$\beta \geqslant 1 - k\left[\exp\left(-m(c-1)^2/2\right) + \frac{1}{c}(1+m)\,e^{m}\right] = 1 - o(1),$$

since $k = e^{o(m)}$ and $c \in (0, 1)$. □

This theorem demonstrates that the space optimal version of ASSIGN will give solutions of expected size roughly $\sqrt{n}$ times that of the optimal solution. For the time optimal version, the factor is even worse at $n^{3/4}$. By comparison, GREEDY gives a solution that is approximately $\log n$ times optimal. The best heuristic for this kind of input appears to be the $O(n^2)$ heuristic of Lai [4]. This gives an expected solution of size at most 6, which is a small constant larger than optimal.

References

[1] D. Avis, "A survey of heuristics for the weighted matching problem", *Network* **13**, 475–494 (1983).

[2] L. Devroye, "Laws of the iterated logarithm for order statistics of uniform spacings", *Annals of Probability* **9**, 860–867.

[3] J.M. Kurtzberg, "On approximation methods for the assignment problem", *JACM* **9**, 419–439 (1962).

[4] C.W. Lai, A heuristic for the assignment problem and related bounds, M.Sc. Thesis, Technical Report 81.20, School of Computer Science, McGill University, 1981.

[5] C.H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice Hall, Englewood Cliffs, NJ, 1982.

[6] D.W. Walkup, "On the expected value of a random assignment problem", *SIAM J. Comp.* **8**, 440–442 (1979).