Angeline Brandt; Jakub Intrator

The assignment problem with three job categories

Časopis pro pěstování matematiky, Vol. 96 (1971), No. 1, 8--11

Persistent URL: http://dml.cz/dmlcz/117707

Časopis pro pěstování matematiky, roč. 96 (1971), Praha

THE ASSIGNMENT PROBLEM WITH THREE JOB CATEGORIES

A. BRANDT, Y. INTRATOR, Rehovot

(Received December 16, 1968)

1. Introduction. The personnel-assignment problem can be formulated as follows: A certain company needs to fill N jobs, and N applicants are available. The value (i.e., the net profit to the company) of each applicant depends upon the job in which he is placed. Assuming these values to be known, the problem is to assign the applicants in such a way that the total value is a maximum.

Frequently there are many identical jobs which demand the same qualifications. Such jobs can be combined into a job category.

In this paper we give a very simple conbinatorial method to solve the assignment problem with three such job categories. This method is very efficient even for big N, the maximum number of operations being roughly $KN \log N$, where K is independent of N.

Let b_j denote the number of jobs grouped in the j^{th} category, v_{ij} the values of the i^{th} applicant when placed in the j^{th} category $(i = 1, 2, \ldots, N)$; $(j = 1, 2, 3)$. Thus $N = b_1 + b_2 + b_3$.

Definitions. *An assignment* s is a function which assigns the i^{th} applicant to the $s(i)^{\text{th}}$ job category, $i = 1, 2, \ldots, N$, $1 \leqq s(i) \leqq 3$. *A feasible assignment* in our problem is an assignment which assigns exactly b_j applicants to the j^{th} category, $j = 1, 2, 3$. *The total value* of a feasible assignment s is

$$v(s) = \sum_{i=1}^{N} v_{i,s(i)}$$

Our assignment problem is, of course, to find a feasible assignment the total value of which is a maximum. There may be, however, several such maximum assignments. To avoid ambiguity we introduce the following "perturbed" values:

$$v^{\varepsilon}_{ij} = v_{ij} + j\varepsilon^i , \quad v^{\varepsilon}(s) = \sum_{i=1}^{N} v^{\varepsilon}_{i,s(i)} .$$

It is easy to see that there is one and only one feasible assignment s_0 whose perturbed-total-value is a maximum *for all sufficiently small positive* ε, i.e, $v^\varepsilon(s_0) > v^\varepsilon(s)$ for all feasible solutions $s \neq s_0$ and all $0 < \varepsilon < \varepsilon_0$. We call s_0 *the maximum assignment.* It is our purpose to compute s_0.

2. The solution of the Problem. For fixed k and l $(1 \leqq k \neq l \leqq 3)$, we compute the differences

$$\Delta_{kl}(i) = v_{ik} - v_{il}\,, \quad i = 1, 2, \ldots, N\,,$$

and arrange the N applicants in a sequence of decreasing $\Delta_{kl}(i)$. Let $p_{kl}(i)$ denote the place of the i^{th} applicant in that sequence. Thus $p_{kl}(i)$ is a permutation such that $p_{kl}(i_1) < p_{kl}(i_2)$ if and only if

$$\text{either} \quad \Delta_{kl}(i_1) > \Delta_{kl}(i_2)$$

$$\text{or} \quad \Delta_{kl}(i_1) = \Delta_{kl}(i_2) \quad \text{and} \quad (k - l)(i_2 - i_1) > 0\,.$$

Note that these rules are designed to ensure *strictly decreasing* order in the *perturbed* differences

$$\Delta^\varepsilon_{kl}(i) = v^\varepsilon_{ik} - v^\varepsilon_{il} = \Delta_{kl}(i) + (k - l)\,\varepsilon^i\,, \quad (0 < \varepsilon < \varepsilon_0)\,.$$

Also note that p_{kl} is the reverse of p_{lk}, namely, $p_{kl}(i) = N + 1 - p_{lk}(i)$.

Proposition 1. *If* $p_{kl}(i) \leqq b_k$ *then* $s_0(i) \neq l$.

In other words, the first b_k applicants in the sequence cannot be assigned, in the maximum assignment, to the l^{th} category. We shall say that they are *labeled as non-l workers.*

Proof. Suppose $s_0(i) = l$. Then there must exist i_1 such that $s_0(i_1) = k$ and $p_{kl}(i_1) > b_k$. Let s_1 be an assignment identical with s_0 except for $s_1(i) = k$ and $s_1(i_1) = l$. Thus

$$v^\varepsilon(s_1) = v^\varepsilon(s_0) + \Delta^\varepsilon_{kl}(i) - \Delta^\varepsilon_{kl}(i_1)\,.$$

But $p_{kl}(i_1) > p_{kl}(i)$ and therefore $\Delta^\varepsilon_{kl}(i) > \Delta^\varepsilon_{kl}(i_1)$. Hence $v^\varepsilon(s_1) > v^\varepsilon(s_0)$, which contradicts our definition of the maximum assignment s_0. Therefore we must have $s_0(i) \neq l$.

Similarly we can show that if $p_{kl}(i) > N - b_l$ then $s_0(i) \neq k$. Thus the last b_l applicants in the p_{kl} sequence are *labeled as non-k workers.*

Our algorithm proceeds as follows: We write down the three sequences (permutations) p_{12}, p_{23} and p_{31}. The first b_k and the last b_l applicants of each sequence p_{kl} are labeled as non-l and non-k workers, respectively. Since each applicant appears in all three sequences, many of them are likely to get two different labels and thus be conclusively assigned. For example, if $p_{12}(i) \leqq b_1$ and $p_{31}(i) > N - b_1$ then by the above proposition $s_0(i) \neq 2$ and $s_0(i) \neq 3$, and hence $s_0(i) = 1$. The i^{th} applicant is then assigned to the first job category, and the problem dimension N is reduced.

In this way we can continue to mark workers off until the problem is either completely solved or reduced to a problem where no applicant has two diffrent labels. This reduced problem is immediately solved by Propositions 2 and 3 below.

Proposition 2. *If no applicant has two different labels then $b_1 = b_2 = b_3$ and every applicant has exactly one label.*

Proof. With no loss of generality we may assume that $b_1 \geqq b_2 \geqq b_3$. Let $v \geqq 0$ be the number of applicants with no label. Using Proposition 1 we get at least b_2 applicants with non-1 labels, at least b_1 with non-2 labels and at least b_1 with non-3 labels. Thus the total number of different labels is at least $2b_1 + b_2$, i.e.,

$$b_1 + b_2 + b_3 - v \geqq 2b_1 + b_2 .$$

This implies $b_3 \geqq b_1 + v \geqq b_3 + v$, which in turn yields $v = 0$ and $b_1 = b_3$. Q.E.D.

Thus, in the reduced problem we have $N = 3b$, $b_j = b$, and, for a fixed $1 \leqq \leqq j \leqq 3$, exactly b applicants are labeled as non-j workers. Let these b applicants be ordered in a sequence of decreasing Δ_{kl} (strictly decreasing $\Delta^{\varepsilon}_{kl}$. (j, k, l) here is any cyclic permutation of $(1, 2, 3)$), and denote by $i_{kl}(p)$ the index of the applicant that appears in the p^{th} place of this sequence. In fact this ordering requires no extra effort, for, by Proposition 1–2, all the non-j labeled applicants appear at the places $b + 1$, $b + 2, \ldots, 2b$ in the sequence p_{kl}, where they are ordered in decreasing Δ_{kl} (strictly decreasing $\Delta^{\varepsilon}_{kl}$); i.e.,

$$i_{kl}(p_{kl}(i) - b) = i\,, \quad b < p_{kl}(i) \leqq 2b\,.$$

Now, clearly, if $s_0(i_{kl}(p_0)) = k$ then $s_0(i_{kl}(p)) = k$ for all $p < p_0$. For if $s_0(i_{kl}(p)) = = l$ we could define another assignment, s_1 identical with s_0 except for $s_1(i_{kl}(p_0)) = l$ and $s_1(i_{kl}(p)) = k$, for which we would have

$$v^{\varepsilon}(s_1) = v^{\varepsilon}(s_0) - \Delta^{\varepsilon}_{kl}(i_{kl}(p_0)) + \Delta^{\varepsilon}_{kl}(i_{kl}(p)) > v^{\varepsilon}(s_0)\,,$$

in contradiction to the definition of s_0. Therefore there must be a certain place, q_{kl} say $(0 \leqq q_{kl} \leqq b)$, such that

$$s_0(i_{kl}(p)) = \begin{cases} k & \text{for} \quad 1 \leqq p \leqq q_{kl} \\ l & \text{for} \quad q_{kl} < p \leqq b\,. \end{cases}$$

For this assignment to be feasible there must be exactly b applicants assigned to the k^{th} category, that is,

$$q_{kl} + (b - q_{jk}) = b\,.$$

This entails $q_{12} = q_{23} = q_{31} = q_0$ (say), and therefore $s_0 = s^{q_0}$, where the assignments s^q $(0 \leqq q \leqq b)$ are defined by

$$s^q(i_{kl}(p)) = \begin{cases} k & \text{for} \quad 1 \leqq p \leqq q\,, \\ l & \text{for} \quad q < p \leqq b\,, \quad (k, l) = (1, 2), (2, 3), (3, 1)\,. \end{cases}$$

All we need to conclude our solution is to determine q_0. This is done by observing that

$$v^\varepsilon(s^q) - v^\varepsilon(s^{q-1}) = C^\varepsilon(q)$$

where

$$C^\varepsilon(q) = \Delta^\varepsilon_{12}(i_{12}(q)) + \Delta^\varepsilon_{23}(i_{23}(q)) + \Delta^\varepsilon_{31}(i_{31}(q)) =$$
$$= C(q) - \varepsilon^{i_{12}(q)} - \varepsilon^{i_{23}(q)} + 2\varepsilon^{i_{31}(q)}\,,$$
$$C(q) = \Delta_{12}(i_{12}(q)) + \Delta_{23}(i_{23}(q)) + \Delta_{31}(i_{31}(q))\,.$$

Since $\Delta^\varepsilon_{kl}(i_{kl}(q))$ is a strictly decreasing function of q, so is also $C^\varepsilon(q)$ and hence q_0 must be the last q for which $C^\varepsilon(q) > 0$ for all sufficiently small positive ε. We have thus proved

Proposition 3. *Let q_0 be the last q for which*

either $C(q) > 0$

or $C(q) = 0,\ i_{31}(q) < i_{12}(q)$ *and* $i_{31}(q) < i_{23}(q)$.

(*Put $q_0 = 0$ if no q satisfies this condition.*) *Then $s_0 = s^{q_0}$.*

This proposition gives us the solution of the reduced problem (where no applicant has two different labels) and thus conclude the solution of the problem.

Number of operations. The above algorithm can be programmed so that the assignment of each applicant takes no more than a fixed number k of operations, k being independent of N. To this one has to add the three sortings needed initially to set up the permutations $P_{12}(i)$, $P_{23}(i)$ and $P_{31}(i)$. Each such sorting can be executed in at most $KN \log N$ operations, K being independent of N. Thus, for big N, the initial sortings constitute the most time consuming part of the algorithm, and the total number of operations is $O(N \log N)$.

Authors' address: The Weizmann Institute of Science, Rehovot, Israel.