Discrete Applied Mathematics 41 (1993) 139–146
North-Holland

On polynomial solvability of the high multiplicity total weighted tardiness problem

Frieda Granot*

Faculty of Commerce and Business Administration, The University of British Columbia, Vancouver, BC, Canada

Jadranka Skorin-Kapov**

W.A. Harriman School for Management and Policy, State University of New York at Stony Brook, Stony Brook, NY, USA

Received 12 December 1989
Revised 23 January 1991

Abstract

Granot, F. and J. Skorin-Kapov, On polynomial solvability of the high multiplicity total weighted tardiness problem, Discrete Applied Mathematics 41 (1993) 139–146.

In a recent paper Hochbaum et al. developed a polynomial algorithm for solving a scheduling problem of minimizing the total weighted tardiness for a large number of unit length jobs which can be partitioned into few sets of jobs with identical due dates and penalty weights. The number of unit jobs in a set is called the multiplicity of that set. The problem was formulated in Hochbaum et al. as an integer quadratic nonseparable transportation problem and solved, in polynomial time, independent of the size of the multiplicities and the due dates but depending on the penalty weights. In this paper we show how to solve the above problem in polynomial time which is independent of the sizes of the weights. The running time of the algorithm depends on the dimension of the problem and only the size of the maximal difference between two consecutive due dates. In the case where the due dates are large, but the size of the maximal difference between two consecutive due dates is polynomially bounded by the dimension of the problem, the algorithm runs in strongly polynomial time.

1. Introduction

In a recent paper by Hochbaum et al. [4] the following scheduling problem was addressed. Many jobs of unit length but with only a few sets of distinct due dates

Correspondence to: Professor F. Granot, Faculty of Commerce and Business Administration, The University of British Columbia, 2053 Main Mall, University Campus, Vancouver, BC, Canada V6T 1Y8.

* Partially supported by Natural Sciences and Engineering Research Council of Canada Grant 5-83998.
** Partially supported by National Science Foundation Grant DDM-8909206.

and penalty weights need to be scheduled on a single machine. The objective is to minimize the total weighted tardiness. In [4], the problem was formulated as an integer quadratic nonseparable transportation problem. It was solved by first solving a related continuous quadratic problem and then a rounding procedure was used to derive the optimal integral solution from the continuous one. In order to obtain the optimal solution for the continuous quadratic programming problem in polynomial running time, independent of the multiplicities and the due dates, a simultaneous approximation algorithm can be used (see Chandrasekaran and Kabadi [1], or in the case of distinct weights, also Granot and Skorin-Kapov [3]).

Recall that the size of a number is the length of its binary description (i.e., the number of bits needed to record a given number in a binary format). In this paper we show how to solve the above problem in time which polynomially depends only on the dimension of the problem and on the size of the maximal difference between consecutive due dates. We suggest two alternatives. Either solve a single related continuous quadratic program and then use a rounding procedure to obtain the required integral solution, or else solve a sequence of linear programming problems where the number of problems solved is bounded by the size of the maximal difference between two consecutive due dates. If the size of the maximal difference between consecutive due dates is polynomially bounded by the dimension of the problem, both alternatives will result with algorithms which run in strongly polynomial time. In other words, the proposed algorithms are strongly polynomial in the case when all due dates are "very big", that is, all clustered close to the sum of all multiplicities.

2. Problem formulation and transformation

Following the notation of Hochbaum et al. [4] consider n sets of unit jobs. Set i includes p_i jobs all having the same due date d_i and penalty weight w_i. We refer to p_i as the multiplicity of type i and denote by $P=\sum_{i=1}^{n} p_i$ the total number of unit jobs. We are seeking an assignment of the unit jobs to the P distinct time intervals $(i-1,i]$, $i=1,\dots,P$ which will minimize the total weighted tardiness. The weighted tardiness of each unit of type i scheduled at interval $(t-1,t]$ is $w_i \max(t-d_i,0)$.

Assume without loss of generality that

$$w_1 \geq w_2 \geq \cdots \geq w_n > 0.$$

Permute the types of jobs so that $d_{\pi(1)} < \cdots < d_{\pi(n)}$ and set $d_{\pi(0)}=0$ and $d_{\pi(n+1)}=P$. Define the ith due-date interval as $(d_{\pi(i)}, d_{\pi(i+1)}]$ for $i=0,\dots,n$. The weight of job j in interval i, denoted by $w_j^{(i)}$, is defined to equal zero if job j is not tardy (i.e., $d_{\pi(i)} < d_j$) and w_j otherwise. Assume without loss of generality that all the due dates are distinct, that is, there are no empty intervals. This since empty intervals can be omitted from further consideration resulting with a reduced number of intervals.

Hochbaum et al. [4] gave the following integer quadratic programming formulation of the total weighted tardiness problem:

$$
\begin{aligned}
\min \quad & \sum_{i=0}^{n} \sum_{j=1}^{n} \left[w_j^{(i)}(d_{\pi(i)} - d_j + \tfrac{1}{2})x_{ij} + w_j^{(i)}\left(\tfrac{1}{2}x_{ij}^2 + x_{ij} \sum_{k \mid w_k^{(i)} \geq w_j^{(i)}, k<j} x_{ik}\right)\right], \\
\text{s.t.} \quad & \sum_{i=0}^{n} x_{ij} = p_j, \qquad j=1,\dots,n, \\
& \sum_{j=1}^{n} x_{ij} = \delta_i, \qquad i=0,\dots,n, \\
& x_{ij} \geq 0 \text{ and integer}, \quad i=0,\dots,n,\ j=1,\dots,n
\end{aligned}
\tag{1}
$$

where $\delta_i = d_{\pi(i+1)} - d_{\pi(i)}$ denotes the length of the ith due-date interval, and x_{ij} is the number of units of type j in interval i.

In this paper we introduce a transformation that will enable us to replace the above nonseparable quadratic integer programming problem (1) with an equivalent separable problem in which the constraint matrix remains totally unimodular.

Theorem 1. *Problem* (1) *is equivalent to a separable quadratic integer programming problem having* $(n^2+5n+2)/2$ *constraints and* $3n(n+1)/2$ *variables. Moreover, the constraint matrix of the equivalent problem is totally unimodular.*

Proof. In order to prove the theorem, we restate problem (1) by distinguishing between tardy and nontardy job units in each interval. Let us denote by y_{ij} (respectively, x_{ij}) the number of nontardy (respectively, tardy) units of type j in interval i and by $w_{ip(j)}$ the weight of the jth tardy type in interval i. Clearly, $w_{ip(1)} \geq w_{ip(2)} \geq \cdots \geq w_{ip(i)} > w_{ip(i+1)} = 0$. Moreover, in interval $(0, d_{\pi(1)}]$ there are no tardy jobs and in interval $(d_{\pi(n)}, P]$ all the jobs are tardy. In interval i, i.e., $(d_{\pi(i)}, d_{\pi(i+1)}]$ tardy types are $\pi(1), \dots, \pi(i)$ and nontardy are $\pi_{(i+1)}, \dots, \pi_{(n)}$. Using this observation, the constraint set can be written as

$$
\begin{aligned}
& \sum_{i=0}^{j-1} y_{i\pi(j)} + \sum_{i=j}^{n} x_{i\pi(j)} = p_{\pi(j)}, \qquad j=1,\dots,n, \\
& \sum_{j=i+1}^{n} y_{i\pi(j)} + \sum_{j=1}^{i} x_{i\pi(j)} = \delta_i, \qquad i=0,\dots,n, \\
& y_{i\pi(j)}, x_{i\pi(j)} \geq 0 \text{ and integer}, \quad i=0,\dots,n,\ j=1,\dots,n.
\end{aligned}
$$

In the objective function only the tardy types, i.e., x-variables will appear. In any interval i, tardy types will be scheduled in a nonincreasing order of their weights, say $(p(1), p(2), \dots, p(i))$. The quadratic part of the objective function of (1) for interval i will thus be given by $\sum_{j=1}^{i} w_{ip(j)}\{\frac{1}{2}x_{ip(j)}^2 + \sum_{k=1}^{j-1} x_{ip(k)}x_{ip(j)}\}$. This expression can be further written as

$$
\sum_{j=1}^{i} w_{ip(j)} \left\{ \tfrac{1}{2}x_{ip(j)}^2 + \sum_{k=1}^{j-1} x_{ip(k)}x_{ip(j)} \right\}
$$

$$= \sum_{j=1}^{i} \left[\sum_{l=j}^{i} (w_{ip(l)} - w_{ip(l+1)}) \right] \left\{ \tfrac{1}{2} x_{ip(j)}^2 + \sum_{k=1}^{j-1} x_{ip(k)} x_{ip(j)} \right\}$$

$$= \sum_{l=1}^{i} \left[\sum_{j=1}^{l} (w_{ip(l)} - w_{ip(l+1)}) \right] \left\{ \tfrac{1}{2} x_{ip(j)}^2 + \sum_{k=1}^{j-1} x_{ip(k)} x_{ip(j)} \right\}$$

$$= \sum_{l=1}^{i} (w_{ip(l)} - w_{ip(l+1)}) \sum_{j=1}^{l} \left\{ \tfrac{1}{2} x_{ip(j)}^2 + \sum_{k=1}^{j-1} x_{ip(k)} x_{ip(j)} \right\}$$

$$= \sum_{l=1}^{i} (w_{ip(l)} - w_{ip(l+1)}) \left\{ \tfrac{1}{2} \sum_{j=1}^{l} x_{ip(j)}^2 + \tfrac{1}{2} \sum_{\{1 \le k \ne j \le l\}} x_{ip(k)} x_{ip(j)} \right\}$$

$$= \sum_{l=1}^{i} (w_{ip(l)} - w_{ip(l+1)}) \tfrac{1}{2} \left[\sum_{t=1}^{l} x_{ip(t)} \right]^2 = \tfrac{1}{2} \sum_{l=1}^{i} (w_{ip(l)} - w_{ip(l+1)}) u_{il}^2.$$

The equivalent separable quadratic programming problem of (1) will thus be

$$\begin{aligned}
\min \quad & \sum_{i=1}^{n} \sum_{j=1}^{i} w_{i\pi(j)} (d_{\pi(i)} - d_{\pi(j)} + \tfrac{1}{2}) x_{i\pi(j)} + \tfrac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{i} (w_{ip(j)} - w_{ip(j+1)}) u_{ij}^2, \\
\text{s.t.} \quad & \sum_{i=0}^{j-1} y_{i\pi(j)} + \sum_{i=j}^{n} x_{i\pi(j)} = p_{\pi(j)}, \quad j = 1, \ldots, n, \\
& \sum_{j=i+1}^{n} y_{i\pi(j)} + \sum_{j=1}^{i} x_{i\pi(j)} = \delta_i, \quad i = 0, \ldots, n, \\
& \sum_{k=1}^{j} x_{ip(k)} - u_{ij} = 0, \quad i = 1, \ldots, n,\ j = i, i-1, \ldots, 1, \\
& y_{i\pi(j)} \ge 0 \text{ and integer}, \quad i = 0, \ldots, n-1,\ j = i+1, \ldots, n, \\
& x_{i\pi(j)} \ge 0 \text{ and integer}, \quad i = 1, \ldots, n,\ j = 1, \ldots, i, \\
& u_{ij} \ge 0 \text{ and integer}, \quad i = 1, \ldots, n,\ j = 1, \ldots, i.
\end{aligned} \tag{2}$$

This problem has $n(n+1)/2$ y-variables $n(n+1)/2$ x-variables and $n(n+1)/2$ u-variables. The number of constraints is $n + (n+1) + n(n+1)/2 = (n^2 + 5n + 2)/2$.

It remains to show that the constraint matrix of (2) is totally unimodular. To that end observe that the constraint matrix of (2) is of the form

$$\tilde{A} = \begin{pmatrix} A_{\mathrm{N}} & A_{\mathrm{T}} & 0 \\ 0 & B & -I \end{pmatrix}$$

where A_{N} (respectively, A_{T}) represents the part of $\tilde{A}$ associated with nontardy (respectively, tardy) types, $-I$ is a negative identity $(n(n+1)/2) \times (n(n+1)/2)$ matrix corresponding to the variables $u_{11}, u_{22}, u_{21}, u_{33}, u_{32}, u_{31}, \ldots, u_{nn}, u_{n(n-1)}, \ldots, u_{n1}$. Matrix B is then an $n(n+1)/2 \times n(n+1)/2$ block diagonal matrix where the ith diagonal block denoted by C_i, $i = 1, \ldots, n$ is an $i \times i$ matrix. We claim that C_i is an interval matrix, that is, a zero one matrix in which the ones in each column are consecutive.

To that end, recall that for interval i the u-variables are ordered as $u_{ii}, u_{ii-1}, \dots, u_{i1}$ and that $u_{ij} = \sum_{k=1}^{j} x_{ip(k)}$, for $j \le i$.

If a variable, say $x_{i\pi(e)}$, is not included in the definition of u_{ij} (i.e., its weight is not among the j largest weights in that interval) then trivially, it is not among the $j-1$ largest weights. Therefore $x_{i\pi(e)}$ is not included in the definition of $u_{ij-1}, \dots, u_{i1}$. This proves the claim since if a variable $x_{i\pi(j)}$ has a zero coefficient in any constraint in C_i it will also have zero coefficients in all other constraints following that one.

Recall also that interval matrices are totally unimodular (see e.g., [6]). Moreover, if $\binom{A_N\ A_T}{0\ \ B}$ is totally unimodular, then so is $\tilde{A}$. In order to show that $\binom{A_N\ A_T}{0\ \ B}$ is totally unimodular we will show that by performing linear operations on its rows, it can be transformed into an interval matrix. This clearly preserves the total unimodularity. Let us denote by $a_1, \dots, a_n$ the first n rows in $\binom{A_N\ A_T}{0\ \ B}$ and by $b_0, \dots, b_n$ the next $n+1$ rows. Further, the rows of $(0, B)$ will be denoted by $c_{11}, c_{21}, c_{22}, c_{31}, c_{32}, c_{33}, \dots, c_{n1}, c_{n2}, \dots, c_{nn}$.

Perform now the following linear operations:

- For $k = n-1, \dots, 1$ add row a_k to rows $a_{k+1}, \dots, a_n$;
- for $i = 1, \dots, n$ add rows b_j, $j = i, \dots, n$ to b_{i-1};
- for $j = 1, \dots, n$ add row c_{j1} to $b_{j+1}, \dots, b_n$;
- add c_{j1} to all rows preceding it in $(0, B)$ for $j = 2, \dots, n$.

The matrix $\binom{A_N\ A_T}{0\ \ B}$ is thus transformed to an interval matrix and therefore the matrix $\tilde{A}$ is totally unimodular. □

Example. For the sake of clarity we calculate the matrices A_N, A_T and B for a simple example. Suppose we have three types of jobs with respective parameters:

$$w_1 = 3, \quad d_1 = 4, \quad p_1 = 3;$$
$$w_2 = 2, \quad d_2 = 1, \quad p_2 = 3;$$
$$w_3 = 1, \quad d_3 = 2, \quad p_3 = 3.$$

Therefore, $P = 9$; $\delta_0 = (0,1]$, $\delta_1 = (1,2]$, $\delta_2 = (2,4]$, $\delta_3 = (4,9]$; and $\pi(1) = 2$, $\pi(2) = 3$, $\pi(3) = 1$.

$$A_N = \begin{pmatrix} 1\,0\,0 & 0\,0 & 0 \\ 0\,1\,0 & 1\,0 & 0 \\ 0\,0\,1 & 0\,1 & 1 \\ & & \\ 1\,1\,1 & 0\,0 & 0 \\ 0\,0\,0 & 1\,1 & 0 \\ 0\,0\,0 & 0\,0 & 1 \\ 0\,0\,0 & 0\,0 & 0 \end{pmatrix}, \qquad A_T = \begin{pmatrix} 1 & 1\,0 & 1\,0\,0 \\ 0 & 0\,1 & 0\,1\,0 \\ 0 & 0\,0 & 0\,0\,1 \\ & & \\ 0 & 0\,0 & 0\,0\,0 \\ 1 & 0\,0 & 0\,0\,0 \\ 0 & 1\,1 & 0\,0\,0 \\ 0 & 0\,0 & 1\,1\,1 \end{pmatrix},$$

$$B = \begin{pmatrix} 1 & 0\,0 & 0\,0\,0 \\ & & \\ 0 & 1\,1 & 0\,0\,0 \\ 0 & 1\,0 & 0\,0\,0 \\ & & \\ 0 & 0\,0 & 1\,1\,1 \\ 0 & 0\,0 & 1\,0\,1 \\ 0 & 0\,0 & 0\,0\,1 \end{pmatrix},$$

with $C_1 = 1$; $C_2 = \binom{11}{10}$, and

$$C_3 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}.$$

C_2 has the given form since among the tardy types within the second interval job 2 ($= \pi(1)$) has bigger weight than job 3 ($= \pi(2)$). In the third interval all the types are tardy and since $\pi(1) = 2$, $\pi(2) = 3$, $\pi(3) = 1$ or $w_{\pi(3)} > w_{\pi(1)} > w_{\pi(2)}$, it follows that $u_{33} = x_{3\pi(1)} + x_{3\pi(2)} + x_{3\pi(3)}$; $u_{32} = x_{3\pi(1)} + x_{3\pi(3)}$; $u_{31} = x_{3\pi(3)}$.

3. Polynomial solvability

After the transformation of problem (1) to problem (2) which is a separable quadratic problem with integral coefficients and totally unimodular constraint matrix, problem (2) can be solved in a number of ways. In the sequel we discuss some possibilities and their complexities.

One approach for solving the total weighted tardiness problem in time polynomial in the dimension of the problem and in the size of the maximal difference between two consecutive due dates is to use the algorithm proposed by Hochbaum and Shanthikumar in [5] for solving an integer nonlinear separable programming problem with a totally unimodular constraint matrix. This involves a solution of $\log_2 U$ linear programs, where U is the maximal upper bound on the variables. In order to assure that the size of the first due date interval $\delta_0 = d_{\pi(1)}$ will not appear in the complexity of the algorithm, the variables in the first due date interval (i.e., $y_{0\pi(1)}, \ldots, y_{0\pi(n)}$) will be expressed as

$$y_{0\pi(j)} = p_{\pi(j)} - \sum_{i=1}^{j-1} y_{i\pi(j)} - \sum_{i=j}^{n} x_{i\pi(j)}, \quad j = 1, \ldots, n.$$

The constraint expressing the sum of variables in the first due date interval will become

$$\sum_{j=1}^{n} \left(p_{\pi(j)} - \sum_{i=1}^{j-1} y_{i\pi(j)} - \sum_{i=j}^{n} x_{i\pi(j)} \right) = \delta_0,$$

or

$$\sum_{j=1}^{n} \sum_{i=1}^{j-1} y_{i\pi(j)} + \sum_{j=1}^{n} \sum_{i=j}^{n} x_{i\pi(j)} = P - \delta_0.$$

The constraints expressing the sum of variables in other due date intervals will not change. Along the same lines as in the proof of Theorem 1 one can check that the constraint matrix of the above problem is still totally unimodular, however, the maximal upper bound on the variables is $U = \max\{d_{\pi(i)} - d_{\pi(i-1)},\ i = 2, \ldots, n+1\}$.

Using Tardos' algorithm [7], the complexity of each linear program in Hochbaum and Shanthikumar's [5] approach will depend only on the dimension of the original problem. If the size of the maximal difference between consecutive due dates is polynomially bounded by the dimension of the problem, the algorithm from [5] is strongly polynomial.

Another approach to solve the total weighted tardiness problem in polynomial time which will depend only on the dimension of the original problem and on the size of U is to first use the simultaneous approximation algorithm for quadratic integer programming problems by Granot and Skorin-Kapov [2] as a preprocessing algorithm. This will result in replacing the penalty weights and due dates with certain integers having sizes polynomially bounded by the dimension of the problem and by the size of the maximal upper bound on the variables without changing the set of optimal solutions for (2).

Recall that for problem (2), all the variables are bounded by the lengths of the due date intervals. We assume that only the first interval (i.e., $(0, d_{\pi(1)}]$) is very big, and that the others are small. Note, however, that the variables $y_{0\pi(1)}, \ldots, y_{0\pi(n)}$ do not appear in the objective function. It can easily be shown that their bounds will not affect the scalar N used in the simultaneous approximation algorithm. Therefore, a scalar $N = 4n^2 U$ can be used.

After the preprocessing is done, one can proceed with the algorithm outlined in [4] to solve the total weighted tardiness problem via the solution of a related continuous quadratic programming problem using Chandrasekaran and Kabadi's algorithm [1] or Granot and Skorin-Kapov's algorithm [3] and then using the rounding procedure of [4]. Note, however, that the requirement that the objective function be strictly convex (i.e., all weights be distinct) in Granot and Skorin-Kapov's algorithm [3] is not necessary. This since for the case where the linear part of the objective function involves only "small" integrals, the transformation of variables proposed in [3] involving the inverse of the objective function matrix in order to homogenize the objective function, is not needed.

Acknowledgement

The authors are grateful to a referee for suggesting a simpler transformation of the objective function than originally used by the authors.

References

[1] R. Chandrasekaran and S. Kabadi, Strongly polynomial algorithm for a class of combinatorial LCPs, Oper. Res. Lett. 6 (1987) 91-92.

[2] F. Granot and J. Skorin-Kapov, On simultaneous approximation in quadratic integer programming, Oper. Res. Lett. 8 (1989) 251-255.

[3] F. Granot and J. Skorin-Kapov, Towards a strongly polynomial algorithm for strictly convex quadratic programs: an extension of Tardos' algorithm, Math. Programming 46 (1990) 225-236.

[4] D.S. Hochbaum, R. Shamir and J.G. Shanthikumar, A polynomial algorithm for an integer quadratic nonseparable transportation problem, Math. Programming 55 (1992) 359-371.

[5] D.S. Hochbaum and J.G. Shanthikumar, Convex separable optimization is not much harder than linear optimization, J. ACM 37 (1990) 843-862.

[6] A. Schrijver, Theory of Linear and Integer Programming (Wiley/Interscience, New York, 1986).

[7] E. Tardos, A strongly polynomial algorithm to solve combinatorial linear programs, Oper. Res. 34 (1986) 250-256.