Journal of the Operations Research
Society of Japan
Vol. 32, No. 3, September 1989

A GENERALIZED VERSION OF ONE MACHINE MAXIMUM LATENESS PROBLEM

Minoru Tada
Ryukoku University
Hiroaki Ishii Toshio Nishida
Osaka University

(Received July 25, 1988)

Abstract In usual scheduling problem, machine speed is considered to be constant. Here we assume that the machine speed is controllable for each job and consider the maximum lateness problem on the single machine. The objective is to determine an optimal schedule and optimal jobwise speed of the machine minimizing the total sum of costs associated with the maximum lateness and jobwise machine speed.

1. Introduction

We consider a generalization of a classical one machine maximum lateness problem. In this problem, machine speed is assumed to be not constant but controllable. That is, machine speed is changeable for each job. The objective is to determine an optimal schedule and optimal jobwise speed of the machine minimizing the total sum of costs associated with the maximum lateness and jobwise machine speed.

In Section 2, we formulate a generalized version of one machine maximum lateness problem. In Section 3, we show how our problem can be solved and propose an algorithm informally. In Section 4, we give an numerical example of this problem.

2. Problem Formulation

First some notations used in this paper are defined. Each job J_j ($1 \leqq j \leqq n$) has the due date d_j and 'standard' processing time t_j (i.e., processing time when machine speed is unit). Let C_j be the completion time and s'_j machine speed for each job J_j. For a convenience sake, we set $s_j = 1/s'_j$. We assume that the cost of machine speed is linear and defined to be $a_j s'_j$, where a_j ($1 \leqq j \leqq n$) is a positive constant. Further, let $d_1 \leqq d_2 \leqq \cdots \leqq d_n$ without any loss of generality. Using these notations, the maximum lateness L_{max} is $\max_j (C_j - d_j)$. We consider the following problem PO.

$$\begin{aligned} &\text{PO: minimize } (a_0 L_{max} + \Sigma_j a_j s'_j) \\ &\text{subject to } s'_j > 0, \end{aligned} \tag{2.1}$$

where a_0 is a positive constant. Let p_j be the actual processing time of each job (i.e., $p_j=t_j s_j$), and set $\bar{a}_j=a_j t_j$. Then the problem P0 is equivalent to the following P1.

(2.2) P1: minimize $(a_0 L_{max}+\Sigma_j \bar{a}_j/p_j)$

Let maximum lateness be t. Then we have the following inequalities,

(2.3) $\sum_{j=1}^{k} p_j-d_k\leqq t \qquad (k=1, 2, \cdots, n)$

since we can assume the optimal processing order of jobs is $J_1\to J_2\to\cdots\to J_n$[1]. Thus, from (2.2) and (2.3), we have the following P2 which is equivalent to P0.

P2: minimize $(a_0 t+\sum_{j=1}^{n} \bar{a}_j/p_j)$

(2.4) subject to $p_j>0$,

$$\sum_{j=1}^{k} p_j\leqq t+d_k \qquad (k=1, 2, \cdots, n)$$

3. Solving P2

To solve the problem P2, we use the Lagrange method. Let u_j $(1\leqq j\leqq n)$ be the Lagrange multiplier. Then P2 is equivalent to the following problem:

(3.1) $\max_{u\geqq 0} \min_p L(p,u)$,

where

(3.2) $$L(p,u)=\sum_{j=1}^{n} \bar{a}_j/p_j+\sum_{j=1}^{n} u_j\left(\sum_{k=1}^{j} p_k-d_j-t\right)$$
$$=\sum_{k=1}^{n}\left(\bar{a}_k/p_k+\sum_{j=k}^{n} u_j p_k\right)-\sum_{j=1}^{n} u_j(d_j+t).$$

Let $L(u)=\min_p L(p,u)$, then the inequality of arithmetic and geometric means

$$\bar{a}_k/p_k+\sum_{j=k}^{n} u_j p_k\geqq 2\sqrt{\bar{a}_k\sum_{j=k}^{n} u_j}$$

leads to

(3.3) $$L(p,u)\geqq 2\sum_{k=1}^{n}\sqrt{\bar{a}_k\sum_{j=k}^{n} u_j}-\sum_{j=1}^{n} u_j d_j-t\sum_{j=1}^{n} u_j=L(u).$$

The equality in (3.3) is obtained, when

(3.4) $$p_j=\sqrt{\bar{a}_j/\sum_{k=j}^{n} u_k}.$$

Further, let $y_k=\sqrt{\sum_{j=k}^{n} u_j}$,

i.e.,

(3.5) $$y_k^2=\sum_{j=k}^{n} u_j,$$

or

$$y_k^2 - y_{k+1}^2 = u_k \geq 0, \qquad (k=1, \cdots, n-1)$$
$$y_n^2 = u_n \geq 0.$$

Using these notations, $L(u)$ becomes the following function of y.

$$\begin{aligned} L(u) &= 2\sum_{k=1}^{n}\sqrt{\bar{a}_k}\, y_k - \sum_{k=1}^{n-1}(y_k^2 - y_{k+1}^2)(d_k+t) - y_n^2(d_n+t) \\ &= 2\sum_{k=1}^{n}\sqrt{\bar{a}_k}\, y_k - y_1^2(d_1+t) - \sum_{k=2}^{n} y_k^2(d_k-d_{k-1}) \\ &= -\sum_{k=2}^{n}(d_k-d_{k-1})\left(y_k - \sqrt{\bar{a}_k}/(d_k-d_{k-1})\right)^2 \\ &\quad -(d_1+t)\left(y_1 - \sqrt{\bar{a}_1}/(d_1+t)\right)^2 \\ &\quad +\sum_{k=2}^{n}\bar{a}_k/(d_k-d_{k-1}) + \bar{a}_1/(d_1+t) \triangleq G(y). \end{aligned} \tag{3.6}$$

Now, $e_k (1 \leq k \leq n)$ is defined as follows:

$$e_k = A_k/B_k, \tag{3.7}$$

where

$$A_k = \sqrt{\bar{a}_k} \qquad \text{for } k=1, 2, \cdots, n$$

$$\begin{aligned} B_1 &= d_1 + t, \\ B_k &= d_k - d_{k-1} \quad \text{for } k=2, \cdots, n \end{aligned} \tag{3.8}$$

Then the equation(3.6) is

$$L(u) = -\sum_{k=1}^{n} B_k (y_k - e_k)^2 + \sum_{k=1}^{n} A_k^2/B_k = G(y). \tag{3.9}$$

We have following theorems.

Theorem 1. (i) If $e_k \geq e_{k+1}$ for $k=1, \cdots, n-1$, then by setting $y_j = e_j$ (for $j=1, \cdots, n$), an optimal value $f(t)$ (i.e., $f(t)=\max_{u\geq 0} L(u)$) is obtained as follows:

$$f(t) = \sum_{k=1}^{n} A_k^2/B_k. \tag{3.10}$$

(ii) If there exists an index i such that $e_i < e_{i+1}$, an optimal solution y giving $f(t)$ satisfies the condition $y_i = y_{i+1}$.

Proof: Since the case (i) is clear from (3.5) and (3.9), in order to show (ii), we prove that y satisfying $y_i > y_{i+1}$ does not give $f(t)$.

(a) $y_i > y_{i+1} \geq e_{i+1}$

As y_i tends to e_{i+1}, the value of $G(\cdot)$ increases. Thus, the solution is obtained by setting $y_i = y_{i+1}$.

(b) $y_i > e_{i+1} \geq y_{i+1}$

As y_i and y_{i+1} tend to e_i and e_{i+1}, respectively, the value of $G(\cdot)$ increases. Similarly, the remaining two cases (c) $e_{i+1} > y_i \geq e_i > y_{i+1}$, (d) $e_i \geq y_i > y_{i+1}$ may be proved. Q.E.D.

Theorem 2. If $e_i<e_{i+1}$ for some i $(2\leq i\leq n-1)$, then setting as $y_i=y_{i+1}$ and $e_i=(A_i+A_{i+1})/(B_i+B_{i+1})$ is valid.

Proof: Since validity of setting as $y_i=y_{i+1}$ is clear from Theorem 1, we prove that $e_i=(A_i+A_{i+1})/(B_i+B_{i+1})$. Now it is sufficient to take out terms associated with only the index i and $i+1$ from (3.6). That is,

$$
\begin{aligned}
&2\sqrt{a_i}\,y_i + 2\sqrt{a_{i+1}}\,y_{i+1} - y_i^2(d_i-d_{i-1}) - y_{i+1}^2(d_{i+1}-d_i)\\
&=-(d_{i+1}-d_{i-1})(y_i-(\sqrt{a_i}+\sqrt{a_{i+1}})/(d_{i+1}-d_{i-1}))^2\\
&\quad+(\sqrt{a_i}+\sqrt{a_{i+1}})^2/(d_{i+1}-d_{i-1})\\
&=-(B_i+B_{i+1})(y_i-(A_i+A_{i+1})/(B_i+B_{i+1}))^2\\
&\quad+(A_i+A_{i+1})^2/(B_i+B_{i+1}).
\end{aligned}
$$

Accordingly, an optimal value $f(t)$ is obtained by setting as $y_i=y_{i+1}$ and $e_i=(A_i+A_{i+1})/(B_i+B_{i+1})$. Q.E.D.

Corollary 1. Theorem 2 holds for $i=1$. That is, if $e_1<e_2$, then we can set $y_1=y_2$ and $e_1=(A_1+A_2)/(B_1+B_2)$.

The above theorems suggest an algorithm to solve the problem P2. First, compute e_j for $j=2, \cdots, n$. If the case (ii) of the theorem 1 occurs (i.e., $e_i<e_{i+1}$ for some i), update y_j, e_j as follows (Otherwise, go to the next step without executing the revision.):

$$
\begin{aligned}
&y_j=y_j,\ e_j=e_j,\ \text{if } j=2, \cdots, i-1,\\
&y_i=y_{i+1},\ e_i=(A_i+A_{i+1})/(B_i+B_{i+1}),\ \text{if } j=i,\\
&y_j=y_{j+1},\ e_j=e_{j+1},\ \text{if } j=i+1, \cdots, n-1,
\end{aligned}
$$

and further, update A_j, B_j as follows:

$$
\begin{aligned}
&A_j=A_j,\ B_j=B_j,\ \text{if } j=1, \cdots, i-1,\\
&A_i=A_i+A_{i+1},\ B_i=B_i+B_{i+1},\ \text{if } j=i,\\
&A_j=A_{j+1},\ B_j=B_{j+1},\ \text{if } j=i+1, \cdots, n-1,\\
&n=n-1.
\end{aligned}
\tag{3.11}
$$

After the above revision, test e_j for the case (i) of the theorem. If the case (ii) is happened again, the same revision is repeated. Since the times of revision is at most $(n-1)$, the case (i) certainly occurs, and then the optimal value $f(t)$ is obtained. Next, by the following step, check whether $e_1\geq e_2$ or not. If $e_1\geq e_2$, that is

$$
t\leq\sqrt{a_1}/e_2 - d_1, \tag{3.12}
$$

then set $y_1=e_1$, $y_2=e_2$, $\cdots$, $y_q=e_q$, where q is an integer that is less than or equal to $(n-1)$. Then the shape of objective function in the range (3.12) is

$$(a_0t + \text{const.} + \bar{a}_1/(d_1+t))=F(t). \tag{3.13}$$

Since the differential of $F(t)$ is

$$dF(t)/dt=a_0 - \bar{a}_1/(d_1+t)^2,$$

we obtain the optimal solution as follows:

$$t=\sqrt{a_1/a_0} - d_1. \tag{3.14}$$

Otherwise (i.e., $e_1<e_2$), then reset e_1 as follows:

$$\begin{aligned} &y_1=y_2,\ e_1=(A_1+A_2)/(B_1+B_2),\\ &y_j=y_{j+1},\ e_j=e_{j+1},\ \text{if } j=2,\ \cdots,\ q. \end{aligned} \tag{3.15}$$

If $e_1<e_2$ is occurs again, then the same revision is repeated. Since $e_1\geq e_2$ is certainly occurs after at most $(q-1)$ revisions, we obtain eventually the objective function $F(t)$ to be minimized is determined. And then, since each optimal value of u_j is decided, the optimal processing time is obtained from (3.4).

Theorem 3. The optimal value y^* of y_1 is equal to $\sqrt{a_0}$. Thus, inequality $e_1\geq e_2$ is equivalent to $\sqrt{a_0}\geq e_2$, in other wards, $e_1=\sqrt{a_0}$.

Proof: First, we prove the case without updating e_1. By definition, $e_1=A_1/(d_1 + t)$. Also, from (3.14), $t=A_1/\sqrt{a_0} - d_1$. Thus,

$$e_1=A_1/(d_1 + A_1/\sqrt{a_0} - d_1)=\sqrt{a_0}.$$

Next, we consider the case after executing the revision (3.15). Then, the shape of the objective function is

$$F(t)=(a_0t + \text{const.} + (A_1+A_2)^2/(d_1+t+B_2)).$$

Setting $dF(t)/dt=0$, we obtain $t=(A_1+A_2)/\sqrt{a_0} - (d_1+B_2)$. Using this value of in the updated expression

$$e_1=(A_1+A_2)/(B_1+B_2)=(A_1+A_2)/(t+d_1+B_2),$$

we obtain $e_1=\sqrt{a_0}$. Q.E.D.

Since details of the solution procedure are complicated, we illustrate it by an example in the next section.

4. An Example

We consider an example in order to show the procedure obtaining the solu-

tion. The number of jobs is five and their triplets $T_j=(t_j,d_j,a_j)$ are as follows:

$$T_1=(5,\ 1,\ 3),\ T_2=(4,\ 4,\ 2),\ T_3=(3,\ 4,\ 5),\ T_4=(3,\ 5,\ 1),\ T_5=(5,\ 9,\ 2).$$

Also, let $a_0=4$. First, e_j $(j=2,\ \cdots,\ 5)$ are calculated from (3.7) and (3.8):

$$e_2=\sqrt{8}/3=0.943,\ e_3=\sqrt{15}/0=\infty,\ e_4=\sqrt{3}/1=1.732,\ e_5=\sqrt{10}/4=0.791.$$

The case (ii) of theorem 1 is happened, since $e_2<e_3$. Accordingly, update y_j, e_j from (3.11) as follows:

$$y_2=y_3,\quad e_2=(\sqrt{8}+\sqrt{15})/(3+0)=2.234,$$

$$y_3=y_4,\quad e_3=e_4=\sqrt{3}=1.732,$$

$$y_4=y_5,\quad e_4=e_5=0.791.$$

(4.1) $$A_2=\sqrt{8}+\sqrt{15},\ B_2=3+0,\ A_3=\sqrt{3},\ B_3=1,\ A_4=\sqrt{10},\ B_4=4.$$

Then, the case (i) occurs, that is, $e_2\geq\cdots\geq e_5$.

Next, check whether $e_1\geq e_2$ or not. From theorem 3, $e_1\geq e_2$ is equivalent to $\sqrt{a_0}\geq e_2$. Thus, we check $\sqrt{a_0}\geq e_2$. Since $\sqrt{a_0}=2<2.234=e_2$, the revision (3.15) is executed:

$$y_1=y_2,\quad e_1=(A_1+A_2)/(B_1+B_2),$$

$$y_2=y_3,\quad e_2=e_3=1.732,$$

$$y_3=y_4,\quad e_3=e_4=0.791.$$

(4.2) $$A_1=\sqrt{15}+\sqrt{8}+\sqrt{15},\ B_1=t+4,\ A_2=\sqrt{3},\ B_2=1,\ A_3=\sqrt{10},\ B_3=4.$$

Now the case (i) occurs since $\sqrt{a_0}=2\geq 1.732=e_2$. Thus, we compute t as follows:

$$t=A_1/\sqrt{a_0}-D_k=(\sqrt{15}+\sqrt{8}+\sqrt{15})/2-4=1.287,$$

where $D_k=B_1-t$. In this case, $D_k=d_3$. Each optimal solution is obtained as follows:

$$y_1=y_2=y_3=e_1=\sqrt{a_0}=2,$$

$$y_4=e_2=1.732,\ y_5=e_3=0.791.$$

Next Figure 1 illustrates flow of our solution procedure for this example.

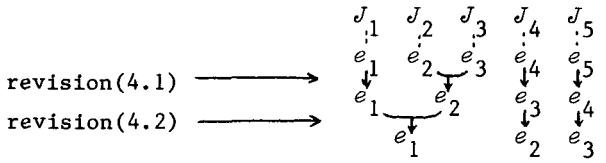

Fig.1 Flow of the solution procedure

Thus, the optimal processing time is obtained from (3.4),

$p_1=\sqrt{15}/2=1.936$, $p_2=\sqrt{8}/2=1.414$, $p_3=\sqrt{15}/2=1.936$,

$p_4=\sqrt{3}/1.732=1$, $p_5=\sqrt{10}/0.791=3.998$.

Accordingly, each optimal speed ($s_j'=t_j/p_j$) is decided as follows:

$s_1'=5/1.936=2.583$, $s_2'=4/1.414=2.829$,

$s_3'=3/1.936=1.550$, $s_4'=3/1=3$, $s_5'=5/3.998=1.251$.

Theorem 4. The solution procedure illustrated by the above example requires $O(n \log n)$ computational time.

Proof: Each computational time of the procedure is shown as follows:

(a) Sorting of due dates d_j: $O(n \log n)$.

(b) Constraction of e_j ($j\geq 2$): $O(n-1)$.

(c) Revision of e_j ($j\geq 1$): $O(n)$,

that is, finding e_2 such that $e_2\leq\sqrt{a_0}$ requires $O(n-1)$ comparisons, and updating of e_j is $O(n)$.

(d) Remaining operations: $O(n)$.

Therefore (a) dominates others and so $O(n \log n)$ computational time is required. Q.E.D.

Reference

[1] W.E.Smith: Various Optimizer for Single Stage Production. *Nav. Res. Log. Quart.*, Vol.3 (1956), 59-66.

Minoru TADA: Faculty of Business Administration, Ryukoku University,
Fukakusa, Fushimi-ku, Kyoto 612, Japan.