Acta Informatica 29:375–382 (1992)

Single machine flow-time scheduling with scheduled maintenance

Chung-Yee Lee[1] and Surya Danusaputro Liman[2]

[1] Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL 32611, USA
[2] Department of Industrial Engineering, Texas Tech University, Lubbock, TX 79409, USA

Received November 13, 1990/January 15, 1992

Abstract. In this paper, we investigate a single machine scheduling problem of minimizing the sum of job flow times subject to scheduled maintenance. We first provide an NP-completeness proof for the problem. This proof is much shorter than the one given in Adiri et al. [1]. The shortest processing time (SPT) sequence is then shown to have a worst case error bound of 2/7. Furthermore, an example is provided to show that the bound is tight. This example also serves as a counter-example to the 1/4 error bound provided in Adiri et al. [1].

1 Introduction

The problem of scheduling n jobs on a single machine to minimize total flow times has been studied for a number of years. It has been shown that the shortest processing time (SPT) sequence, which iteratively assigns the job with the smallest processing time among all yet-to-be-scheduled jobs to the earliest available machines, gives an optimal schedule (see for example, Baker [2], Conway et al. [3], Smith [4]).

Adiri et al. [1] investigate a similar problem except that the machine experiences a single breakdown during the course of processing the jobs. The particular job that happens to be processed when the breakdown occurs must be restarted. Otherwise, preemption is not allowed. They look at both the stochastic and deterministic version of the problem. For the deterministic case, they show that the problem is NP-complete and that the SPT sequence has a relative error bound of 1/4. The SPT sequence processes the jobs in a nondecreasing order of processing times. When the machine breaks down, the job that is currently being processed will have to be reprocessed while the rest of the sequence will remain unchanged.

The problem discussed in this paper is the same as that in Adiri et al. [1] for the deterministic model, although we feel that "scheduled maintenance" is a more appropriate term for the deterministic version of the problem.

In this paper, we will first provide a shorter NP-completeness proof of the single-machine problem. We will then show that the worst case error bound for the SPT sequence is 2/7. Furthermore, we will provide an example to show that the bound is tight which also serves as a counter-example to the 1/4 error bound provided in Adiri et al. [1].

Let J_i denote job i and let p_i be the corresponding processing time for $i = 1, 2, \ldots, n$. Also let R and L denote the starting time and the length of maintenance respectively. Define $|X|$ as the cardinality of set X, and $F(S)$ as the total job flow times of schedule S

2 NP-completeness of the single-machine problem

We now provide a simpler NP-completeness proof for the single-machine case than the one provided by Adiri et al. [1]. We will do so by reducing the even-odd partition problem, which is known to be NP-complete (Garey et al. [5]), in polynomial time into our problem.

The even-odd partition problem:

Given $n \in Z^+$ *and a set* $X = \{x_1, x_2, \ldots, x_{2n}\}$ *of positive integers, where* $x_i < x_{i+1}$ *for* $1 \leqq i < 2n$, *does there exist a partition of* X *into subsets* X_1 *and* X_2 *such that* $\sum_{x \in X_1} x = \sum_{x \in X_2} x$ *and such that for each* i, $1 \leqq i \leqq n$, X_1 *(and hence* X_2*) contains exactly one of* $\{x_{2i-1}, x_{2i}\}$?

The corresponding single-machine problem can be constructed as follows:

Number of jobs: $2n+1$

Processing times: $p_i = M + x_i,\ i = 1, \ldots, 2n$

$p_{2n+1} = P$

Maintenance time: $R = nM + Z$

Length of maintenance: $L = M$

Sum of flow times:
$$y = \sum_{i=1}^{n} (n-i+1)(x_{2i-1} + x_{2i}) + 2n(n+1)M + nZ + \{2(nM+Z)+M\} + P$$

where $Z = \left(\sum_{i=1}^{2n} x_i\right)\Big/2$, $M > 2nZ$, and

$$P > \sum_{i=1}^{n} (n-i+1)(x_{2i-1} + x_{2i}) + 2n(n+1)M + nZ + \{2(nM+Z)+M\}.$$

Note that $2P > y$.

Let sets B and A denote the sets of jobs completed before and after the breakdown respectively. Consider sets B and A' where A' is the set of jobs completed after the breakdown excluding the last job (see Fig. 1).

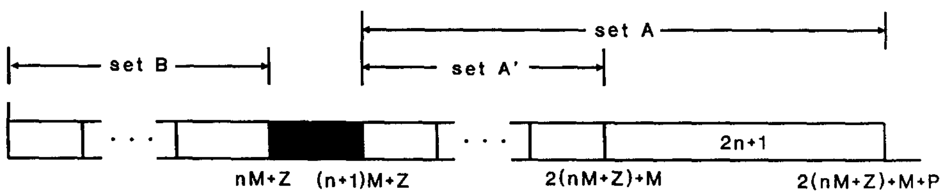

Fig. 1. Schedule S^*

Question. *Does there exist a non-preemptive schedule S^* such that the flow time of the last job in set B is not greater than R and the total flow times of all jobs in S^*, denoted by $F(S^*)$, is no more than y?*

Remark. Note that in the above instance, $0<p_1<p_2<\ldots<p_{2n+1}$.

Lemma 1 *If there exists a solution to the even-odd partition problem, then there exists a schedule S^* for the breakdown problem with total flow times $F(S^*)=y$.*

Proof. Let sets X_1 and X_2 be the solution to the even-odd partition problem. We can construct a schedule S^* of the jobs as shown in Fig. 1, where if x_i is in set X_1 (set X_2), then job i is in set B (resp. set A') and all the jobs in B and A' are in SPT.

Since the solution to the even-odd partition exists, we have

$$\sum_{i\in X_1} x_i = \sum_{i\in X_2} x_i = Z.$$

This implies there exists sets B and A' consisting of the first $2n$ jobs such that

$$\sum_{i\in B} p_i = nM + \sum_{i\in X_1} x_i = nM + Z$$

$$\sum_{i\in A'} p_i = nM + \sum_{i\in X_2} x_i = nM + Z.$$

Let $p_{(i)}$ be the processing time of the i^{th} job in set A'. The flow time of job (i) is given by $\{(n+1)M+Z+\sum_{j\leq i} p_{(j)}\}$. The sum of flow times of jobs in set A' is therefore given by

$$\sum_{j\in A'} C_j(S^*) = n\{(n+1)M+Z\} + \sum_{i=1}^{n} (n-i+1)p_{(i)}.$$

Hence, we obtain

$$\begin{aligned}
F(S^*) &= \sum_{j\in B} C_j(S^*) + \sum_{j\in A'} C_j(S^*) + C_{2n+1}(S^*) \\
&= \sum_{i=1}^{n} (n-i+1)(p_{2i-1}+p_{2i}) + n\{(n+1)M+Z\} + \{(2n+1)M+2Z+p_{2n+1}\} \\
&= \sum_{i=1}^{n} (n-i+1)\{(M+x_{2i-1})+(M+x_{2i})\} \\
&\quad + n\{(n+1)M+Z\} + \{2(nM+Z)+M\} + P \\
&= \sum_{i=1}^{n} (n-i+1)(x_{2i-1}+x_{2i}) + 2n(n+1)M + nZ + \{2(nM+Z)+M\} + P \\
&= y. \quad \text{Q.E.D.}
\end{aligned}$$

Lemma 2 *Let S^* be a feasible schedule for the problem with $F(S^*)\leq y$, then in S^*, the following must be true:*

(1) J_{2n+1} is processed last among all jobs,
(2) $|B|=|A'|=n$,
(3) exactly one of jobs $2i-1$ or $2i$ ($i=1, \ldots, n$) is in B and the other is in A' and $\sum_{j\in B} p_j = \sum_{j\in A'} p_j = nM+Z$ (or equivalently, $\sum_{j\in B} x_j = \sum_{j\in A'} x_j = Z$).

Proof. Part (1) It can be easily checked that $p_{2n+1}>R$. Hence J_{2n+1} must be processed after maintenance. Now, suppose J_{2n+1} is not processed last. This implies that there is another job, say J_j, that is processed after J_{2n+1}. Then

$$\begin{aligned} F(S^*) &\geqq p_{2n+1}+(p_{2n+1}+p_j)\\ &=2p_{2n+1}+p_j\\ &=2P+p_j\\ &>y. \end{aligned}$$

Hence, for $F(S^*)\leqq y$, J_{2n+1} has to be processed last (after set A'). From now on we will only consider sets B and A' which contain the first $2n$ jobs.

Part (2) By definition, $R=nM+Z$ and $p_i>M$ for all J_i. Hence set B cannot contain more than n jobs. Consider a feasible schedule S' whereby $|B|=n'\leqq n-1$ and $|A'|=2n-n'=n''\geqq n+1$. Let $p_{[i]}$ be the processing time of the i^{th} job in schedule S'. Then

$$\begin{aligned} F(S') &= \sum_{i=1}^{n'}(n'-i+1)p_{[i]}+\sum_{i=1}^{n''}(n''-i+1)p_{[n'+i]}+(n'')\{(n+1)M+Z\}+C_{2n+1}(S')\\ &> \frac{n'(n'+1)}{2}M+\frac{n''(n''+1)}{2}M+n''\{(n+1)M+Z\}\\ &\quad +(n+1)M+Z+n''M+P\\ &\geqq n(n+1)M+M+(n+1)\{(n+1)M+Z\}+(n+1)M+Z+(n+1)M+P\\ &=n(n+1)M+M+n\{(n+1)M+Z\}+\{(n+1)M+Z\}+2(n+1)M+Z+P\\ &=2n(n+1)M+nZ+2(nM+Z)+(n+4)M+P\\ &=2n(n+1)M+nZ+\{2(nM+Z)+M\}+P+(n+3)M\\ &>y \quad \text{since } \sum_{i=1}^{n}(n-i+1)(x_{2i-1}+x_{2i})\leqq 2nZ<M<(n+3)M. \end{aligned}$$

Therefore, in order to have $F(S)\leqq y$, we must have $|B|=|A'|=n$.

Part (3) By part (2), we know that in schedule S^*, both sets B and A' contain n jobs.

$$\begin{aligned} F(S^*) &= \sum_{i\in B} C_i(S^*)+\sum_{i\in A'} C_i(S^*)+C_{2n+1}(S^*)\\ &= n\,p_{[1]}+(n-1)\,p_{[2]}+\ldots+p_{[n]}+n(nM+Z+M)\\ &\quad +n\,p_{[n+1]}+(n-1)\,p_{[n+2]}+\ldots+p_{[2n]}\\ &\quad +\{(nM+Z)+M+(p_{[n+1]}+p_{[n+2]}+\ldots+p_{[2n]})+P\}\\ &= n(x_{[1]})+(n-1)(x_{[2]})+\ldots+x_{[n]}\\ &\quad +n(x_{[n+1]})+(n-1)(x_{[n+2]})+\ldots+x_{[2n]}+(x_{[n+1]}+x_{[n+2]}+\ldots+x_{[2n]})\\ &\quad +n\{(n+1)M+Z\}+n(n+1)M+\{2nM+Z+M+P\}\\ &= n(x_{[1]})+(n-1)(x_{[2]})+\ldots+x_{[n]}+\mathrm{n}(x_{[n+1]})+(n-1)(x_{[n+2]})+\ldots+x_{[2n]}\\ &\quad +x_{[n+1]}+x_{[n+2]}+\ldots+x_{[2n]}+(2n^2+4n+1)M+(n+1)Z+P. \end{aligned}$$

For

$$F(S^*) \leqq y = \sum_{i=1}^{n} (n-i+1)(x_{2i-1}+x_{2i}) + 2n(n+1)M + nZ + \{2(nM+Z)+M\} + P,$$

we must have

$$\begin{aligned}(1)\qquad & n(x_{[1]}) + (n-1)(x_{[2]}) + \ldots + x_{[n]} \\ & + n(x_{[n+1]}) + (n-1)(x_{[n+2]}) + \ldots + x_{[2n]} \\ & + x_{[n+1]} + x_{[n+2]} + \ldots + x_{[2n]} \\ & \leqq \sum_{i=1}^{n} (n-i+1)(x_{2i-1}+x_{2i}) + Z.\end{aligned}$$

For notational convenience, let $A = n(x_{[1]}) + (n-1)(x_{[2]}) + \ldots + x_{[n]}$ and $B = n(x_{[n+1]}) + (n-1)(x_{[n+2]}) + \ldots + x_{[2n]}$. Since the coefficients of $x_{[i]}$'s in A and B are both decreasing, if we assign the two smallest x_i's$(=x_1$ or $x_2)$ to either the first or the $(n+1)^{\text{th}}$ positions, the next two smallest x_i's$(=x_3$ or $x_4)$ to either the second or the $(n+2)^{\text{th}}$ positions, etc., the following *minimum* cost is obtained

$$\begin{aligned}(2)\qquad F_{\min} &= \sum_{i=1}^{n} (n-i+1)(x_{2i-1}+x_{2i}) \\ &\leqq n(x_{[1]}) + (n-1)(x_{[2]}) + \ldots + x_{[n]} \\ &\quad + n(x_{[n+1]}) + (n-1)(x_{[n+2]}) + \ldots + x_{[2n]}.\end{aligned}$$

Therefore, we must have $x_{[n+1]} + \ldots + x_{[2n]} \leqq Z$. However, a feasible schedule implies that $x_{[1]} + \ldots + x_{[n]} \leqq Z$. Hence, $x_{[n+1]} + \ldots + x_{[2n]} = Z$ since $x_{[1]} + \ldots + x_{[2n]} = 2Z$. Combining with Eqs. (1) and (2), we have

$$\begin{aligned}(3)\qquad \sum_{i=1}^{n} (n-i+1)(x_{2i-1}+x_{2i}) &= n(x_{[1]}) + (n-1)(x_{[2]}) + \ldots + x_{[n]} \\ &\quad + n(x_{[n+1]}) + (n-1)(x_{[n+2]}) + \ldots + x_{[2n]}.\end{aligned}$$

Namely, the two smallest x_i's$(=x_1$ or $x_2)$ have to be assigned to either the first or the $(n+1)^{\text{th}}$ positions, the next two smallest x_i's$(=x_3$ or $x_4)$ to either the second or the $(n+2)^{\text{th}}$ positions, etc. Q.E.D.

Remark. Part (3) in Lemma 2 implies that there exists an even-odd partition. Combining Lemma 2 with Lemma 1 we have shown the following theorem.

Theorem 1 *The single-machine flow-time scheduling problem with a scheduled maintenance is* NP-*complete.*

3 Error bound for the SPT heuristic

We will now prove that SPT algorithm has a tight error bound of 2/7.

Theorem 2 SPT *has a tight error bound of* 2/7.

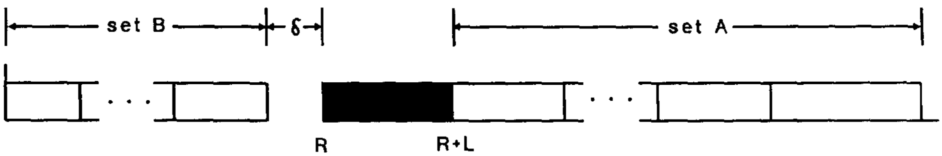

Fig. 2. Schedule S

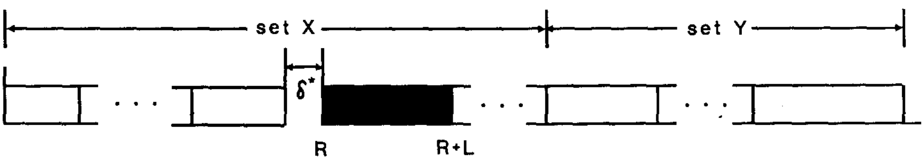

Fig. 3. Schedule S^*

Proof. Consider a schedule S generated by the SPT sequence (see Fig. 2) and an optimal schedule S^* (see Fig. 3), where B and A are the sets of jobs that are scheduled before and after the maintenance in S respectively. Also X is the set of the first $|B|$ jobs in S^*. Hence $|X|=|B|$ and $|Y|=|A|$, where Y is the set of the remaining jobs in S^*. Let δ be the difference between R and the completion time of the last job in B. Namely δ is the gap between B and R. Similarly, let δ^* be the gap between R and the completion time of the last job that is scheduling before the maintenance in schedule S^* (see Fig. 3).

Suppose that $|A|=|Y|=0$, then the problem becomes unconstrained and the SPT sequence is optimal. Furthermore, if $|B|=|X|=0$, then $p_1<R$. In this case, the SPT sequence is again optimal since all jobs have to be processed after the maintenance. Hence, we will only discuss the case with $|A|=|Y|\geqq 1$ and $|B|=|X|\geqq 1$.

Define $F_V(\sigma)$ be the sum of flow times of set V in schedule σ. First note that $(\delta-\delta^*)\geqq 0$. Otherwise, we would have $F(S)=F_A(S)+F_B(S)<F_Y(S^*)+F_B(S)\leqq F(S^*)$ since always $F_B(S)\leqq F_X(S^*)$.

Note also that $\sum_{j\in X} p_j \geqq \sum_{j\in B} p_j$ and all jobs in set Y are completed after the maintenance since set B contains the smallest $|B|$ jobs. Let C_i and C_i^* be the completion times of the first job in set A and Y respectively. We can easily see that $C_i\leqq C_i^*+(\delta-\delta^*)$. Hence

$$F_A(S)\leqq F_Y(S^*)+|Y|(\delta-\delta^*)$$

However, $F_B(S)\leqq F_X(S^*)-(\delta-\delta^*)$, since $\sum_{j\in X} p_j\geqq \sum_{j\in B} p_j+(\delta-\delta^*)$ and set B is in SPT order. We obtain

$$\begin{aligned}F(S)&=F_B(S)+F_A(S)\\&\leqq F_X(S^*)+F_Y(S^*)+(|Y|-1)(\delta-\delta^*)\\&=F(S^*)+(|Y|-1)(\delta-\delta^*).\end{aligned}\tag{4}$$

Since $F(S^*)=F_B(S^*)+F_A(S^*)$ and note that all $p_j \geqq \delta \geqq (\delta-\delta^*)$ for all $J_j \in A$, we have

$$F_A(S^*) \geqq \left\{\frac{|A|(|A|+1)}{2}\right\}(\delta-\delta^*)$$
$$=\left\{\frac{|Y|(|Y|+1)}{2}\right\}(\delta-\delta^*).$$

If SPT is not optimal then set B and X are different, hence there exists a $J_j \in B$ which is processed after the maintenance by the optimal schedule. Then $F_B(S^*) \geqq F_{\{j\}}(S^*) \geqq R+L \geqq \delta-\delta^*$. Hence we have

$$F(S^*) \geqq \left\{\frac{|Y|(|Y|+1)}{2}+1\right\}(\delta-\delta^*). \tag{5}$$

Combining Eqs. (4) and (5), we obtain

$$\varepsilon=\frac{F(S)-F(S^*)}{F(S^*)} \leqq \frac{2(|Y|-1)}{|Y|(|Y|+1)+2}.$$

If $|Y|=1$, $\varepsilon=0$ and S is optimal. For $|Y|=2$, 3, 4, and 5, $\varepsilon=1/4$, $2/7$, $3/11$, and $1/4$ respectively. It can be checked that $\varepsilon<1/4$ for $|Y|>5$. Therefore, SPT algorithm has a error bound of $2/7$ which is greater than $1/4$.

To prove that the error is attainable, consider the following example:

Table 1. Example

Job:	j	1	2	3	4
Processing time:	p_j	1	M	M	M

$R=M$ and $L=1$, where M is a very large number.

Schedule S obtained using SPT is:

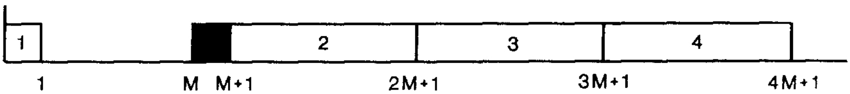

Fig. 4. Schedule S

$$F(S)=9M+4.$$

The optimal schedule S^* is:

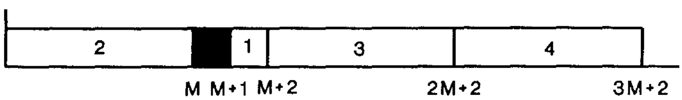

Fig. 5. Schedule S^*

$$F(S^*)=7M+6$$

$\varepsilon=[(9M+4)-(7M+6)]/(7M+6) \approx 2/7$ as M approaches infinity. Hence the error bound is tight. Q.E.D.

4 Conclusion

In this paper, a shorter NP-completeness proof of the single-machine deterministic maintenance problem than the one proposed by Adiri et al. [1] has been provided. SPT sequence has been shown to have a tight relative error bound of 2/7 for the single machine problem. The next natural extension of the problem will be to look at the m-machine problem and generalize the performance of the SPT heuristic for this particular case.

Acknowledgement. We would like to express our appreciation to Professor A.H.G. Rinnooy Kan for his encouragement. We are also grateful to an anonymous referee for the helpful comments.

References

1. Adiri I, Bruno J, Frostig E, Rinnooy Kan AHG: Single machine flow-time scheduling with a single breakdown. Acta Inf. **26**, 679–696 (1989)
2. Conway RW, Maxwell WL, Miller LW: Theory of scheduling. Reading, MA: Addison-Wesley 1967
3. Garey MR, Tarjan RE, Wilfong GT: One-processor scheduling with symmetric earliness and tardiness penalties. Math. Oper. Res. **13**, 330–348 (1988)
4. Smith WE: Various optimizers for single stage production. Naval Res. Log. Qu. **3**, 59–66 (1956)
5. Garey MR, Tarjan RE, Wilfong GT: One-processor scheduling with symmetric earliness and tardiness penalties. Math. Oper. Res. **13**, 330–348 (1988)