NP-hardness of the single-variable-resource scheduling problem to minimize the total weighted completion time

J.J. Yuan[1,2], **T.C.E. Cheng**[2*] *and* **C.T. Ng**[2]

[1]Department of Mathematics, Zhengzhou University,
Zhengzhou, Henan 450052, People's Republic of China
[2]Department of Logistics, The Hong Kong Polytechnic University,
Hung Hom, Kowloon, Hong Kong, People's Republic of China

Abstract: Baker and Nuttle [1] studied the following single-variable-resource scheduling problem: sequencing n jobs for processing by a single resource to minimize a function of job completion times, when the availability of the resource varies over time. When the objective function to be minimized is the total weighted completion time, Baker and Nuttle conjectured that the problem is NP-hard. We show in this note that the conjecture is true.

1 Introduction

Consider the problem of sequencing a set $N = \{1, 2, ..., n\}$ of jobs to be processed using a single resource, where the availability of the resource varies over time. This problem, known as single-variable-resource scheduling, was studied in Baker and Nuttle [1].

For a time instant $t \geq 0$, we denote by $r(t)$ the resource available at time t. For each j with $1 \leq j \leq n$, we denote by $p_j > 0$ the resource requirement of job j. Once p_j units of resource have been applied to job j, the job is considered complete. We assume that all the jobs are released at time 0, the processing of each pair of distinct jobs does not overlap, the processing of every job is continuous (i.e., no preemption is allowed), and there is no idle time between the processing of consecutive jobs. So, a schedule π of the jobs can be specified by a sequence

$$\pi = (\pi(1), \pi(2), ..., \pi(n)),$$

---

[*] Corresponding author

where $\pi(i)$ is the job in the i-th position of π. It was shown in [1] that, under schedule π, the completion time of job $\pi(j)$ can be expressed as a function of the form

$$C_{\pi(j)}(\pi) = f(p_{\pi(1)} + p_{\pi(2)} + ... + p_{\pi(j)}).$$

For the purpose of this paper, we only consider $C_{\pi(j)}$ in the case that there are positive numbers X, a and b such that

$$r(t) = \begin{cases} a, & \text{if } 0 \leq t < X, \\ b, & \text{if } t \geq X. \end{cases}$$

In this case, the completion time of job $\pi(j)$ can be determined by the function

$$C_{\pi(j)} = \begin{cases} P_{\pi(j)}/a, & \text{if } P_{\pi(j)} \leq aX, \\ X + (P_{\pi(j)} - aX)/b, & \text{if } P_{\pi(j)} > aX, \end{cases}$$

where

$$P_{\pi(j)} = p_{\pi(1)} + p_{\pi(2)} + ... + p_{\pi(j)}.$$

For the above problem, when the objective function to be minimized is the total weighted completion time, i.e., $\sum_{1 \leq j \leq n} w_j C_j(\pi)$, Baker and Nuttle [1] conjectured that the problem, denoted by $1^*||\sum w_j C_j$, is NP-hard. It was hinted in Baker and Smith [2] that the computational complexity of $1^*||\sum w_j C_j$ is still open up to now.

We show in this note that Baker and Nuttle's conjecture is true, i.e., $1^*||\sum w_j C_j$ is NP-hard. All the numbers appearing in this note are assumed to be non-negative and integral. So, the result of this note still holds under the variable assumptions in [1].

2 NP-hardness Proof

The standard single-machine total weighted tardiness problem is denoted by $1||\sum w_j T_j$, where each job j has a due date d_j, and under a schedule π, the tardiness of job j is defined by

$$T_j(\pi) = \max\{0, C_j(\pi) - d_j\}.$$

The decision version of $1||\sum w_j T_j$ is denoted by $1||\sum w_j T_j \leq Y$.

By Yuan [3], the scheduling problem $1|d_j = d|\sum w_j T_j$ is NP-hard. In fact, it was shown in [3] that the decision version $1|d_j = d|\sum w_j T_j \leq Y$ is NP-complete.

Theorem $1^*||\sum w_j C_j$ is NP-hard.

Proof Let us be given an instance I of $1|d_j = d|\sum w_j T_j \leq Y$:

$$(p_1, , ..., p_n; w_1, ..., w_n; d; Y),$$

where we have n jobs $J_1, ..., J_n$; $p_j \geq 1$ is the processing time of job J_j, $1 \leq j \leq n$; $w_j \geq 1$ is the weight of job J_j, $1 \leq j \leq n$; $d < p_1 + p_2 + ... + p_n$ is the common due date of the jobs; and $Y > 0$ is the threshold value for the total weighted tardiness of the jobs. The decision problem asks if there is a schedule σ for the n jobs such that $\sum_{1 \leq j \leq n} w_j T_j(\sigma) \leq Y$.

We first define $\Delta = (w_1 + w_2 + ... + w_n)(p_1 + p_2 + ... + p_n)$. Then we construct an instance I^* of the single-variable-resource scheduling problem $1^*|| \sum w_j C_j \leq Q$ as follows.

- n jobs $J'_1, J'_2, ..., J'_n$ with J'_j, $1 \leq j \leq n$, corresponding to job J_j in I.

- The resource requirement of job J'_j is defined by $p'_j = 2\Delta p_j$, $1 \leq j \leq n$, and the weight of job J'_j is defined as $w'_j = w_j$, $1 \leq j \leq n$.

- The resource availability function is defined as

$$r(t) = \begin{cases} 2\Delta, & \text{if } 0 \leq t \leq d, \\ 1, & \text{if } t > d. \end{cases}$$

- Threshold value is defined as $Q = \Delta(2Y + 1)$.

- The decision problem asks if there is a schedule π such that $\sum_{1 \leq j \leq n} w'_j C'_j(\pi) \leq Q$, where $C'_j(\pi)$ is the completion time of job J'_j under π.

For any permutation h of $(1, 2, ..., n)$, h can be thought of as a schedule for both I and I^*. By the definition of $r(t)$, it can be observed that, under schedule h, the completion time of job J'_j is

$$C'_j(h) = \begin{cases} C_j(h), & \text{if } C_j(h) \leq d, \\ d + 2\Delta T_j(h), & \text{if } C_j(h) > d, \end{cases}$$

and so,

$$2\Delta T_j(h) \leq C'_j(h) \leq C_j(h) + 2\Delta T_j(h).$$

By noting that $\Delta \geq \sum_{1 \leq j \leq n} w_j C_j(h)$, one can observe that

$$(1): \quad \sum_{1 \leq j \leq n} w'_j C'_j(h) \leq \Delta + 2\Delta \sum_{1 \leq j \leq n} w_j T_j(h)$$

and

$$(2): \quad 2\Delta \sum_{1 \leq j \leq n} w_j T_j(h) \leq \sum_{1 \leq j \leq n} w'_j C'_j(h).$$

From (1), if $\sum_{1 \leq j \leq n} w_j T_j(h) \leq Y$, then we must have

$$\sum_{1 \leq j \leq n} w'_j C'_j(h) \leq \Delta + 2\Delta Y = Q.$$

From (2), if $\sum_{1 \leq j \leq n} w'_j C'_j(h) \leq Q = \Delta(2Y + 1)$, we must have

$$\sum_{1 \leq j \leq n} w_j T_j(h) \leq Y + 1/2,$$

and by the integrality of the data, we further have

$$\sum_{1 \le j \le n} w_j T_j(h) \le Y.$$

It follows that instance I has a feasible solution if and only if instance I^* has a feasible solution. The result follows. $\square$

Remark: By [3], $1|d_j = d|\sum w_j T_j$ is NP-hard in the ordinary sense. So, we have proved that $1^*||\sum w_j C_j$ is NP-hard. It is still open if the problem $1^*||\sum w_j C_j$ is NP-hard in the strong sense. The method presented in this note does not work for the strong NP-hardness of $1^*||\sum w_j C_j$.

Acknowledgements

This research was supported in part by The Hong Kong Polytechnic University under grant number S818. The first author was also supported in part by the National Natural Science Foundation of China under grant number 10371112.

References

[1] K.R. Baker and H.L.W. Nuttle, Sequencing independent jobs with a single resource, *Naval Research Logistics Quarterly*, **27**(1980), 499-510.

[2] K.R. Baker and J.C. Smith, A multiple-criterion model for machine scheduling. *Journal of Scheduling*, **6**(2003), 7-16.

[3] J.J. Yuan, The NP-hardness of the single machine common due date weighted tardiness problem, *Systems Science and Mathematical Sciences*, **5**(1992), 328-333.