This article was downloaded by: [216.73.216.182] On: 04 June 2026, At: 01:53
Publisher: Institute for Operations Research and the Management Sciences (INFORMS)
INFORMS is located in Maryland, USA

Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Note on “Parallel Machine Scheduling with Batch Setup Times”

Scott Webster,

To cite this article:
Scott Webster, (1998) Note on “Parallel Machine Scheduling with Batch Setup Times”. Operations Research 46(3):423-423. https://doi.org/10.1287/opre.46.3.423

Full terms and conditions of use: https://pubsonline.informs.org/Publications/Librarians-Portal/PubsOnLine-Terms-and-Conditions

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact permissions@informs.org.

The Publisher does not warrant or guarantee the article’s accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

© 1998 INFORMS

Please scroll down for article—it is on subsequent pages

With 12,500 members from nearly 90 countries, INFORMS is the largest international association of operations research (O.R.) and analytics professionals and students. INFORMS provides unique networking and learning opportunities for individual professionals, and organizations of all types and sizes, to better understand and use O.R. and analytics tools and methods to transform strategic visions and achieve better outcomes.
For more information on INFORMS, its publications, membership, or meetings visit http://www.informs.org

TECHNICAL NOTES

NOTE ON “PARALLEL MACHINE SCHEDULING WITH BATCH SETUP TIMES”

SCOTT WEBSTER

Syracuse University, Syracuse, New York

(Received June 1995; revisions received May 1996; accepted July 1996)

Cheng and Chen (1994) use a high-multiplicity encoding scheme to prove binary *NP*-hardness of a scheduling problem. From this they infer a similar result for a well-known, more general problem. We explain that, although their initial proof is correct, their inference about the more general problem is not.

Cheng and Chen (1994) consider the problem of scheduling jobs on two identical parallel processors to minimize total completion time. Each job belongs to a *batch* where jobs from the same batch share a setup. The authors show that the problem is binary *NP*-hard when all job processing times are equal. The result is an important contribution because it addresses a long-standing question in the literature. However, the authors use this complexity result to conclude that a more general problem with arbitrary job processing times is binary *NP*-hard. The purpose of this note is to point out an error in this conclusion.

Following the notation of Cheng and Chen, let $B =$ number of batches, $N_i =$ number of jobs in batch $i \in [1, B]$, $p_{ij} =$ processing time of job $j \in [1, N_i]$ in batch $i \in [1, B]$, $S_i =$ setup time for batch $i \in [1, B]$, $N = \sum_i N_i =$ total number of jobs. All parameters are positive integers.

For a given schedule, let C_{ij} denote the completion time of job j in batch i. The objective is to find a feasible schedule of the N jobs on two identical parallel machines that minimizes $f = \sum_{ij} C_{ij}$, which is the total completion time. Let us denote this problem as **P2TCT1**. The special case of **P2TCT1**, in which all jobs have the same processing time (i.e., $p_{ij} = p$ for all i and j), is denoted **P2TCT2**. Cheng and Chen show that **P2TCT2** is binary *NP*-hard, and they conclude that this implies that **P2TCT1** is binary *NP*-hard.

We will demonstrate that a generalization of a known *NP*-hard problem is *not necessarily NP*-hard and, more specifically, we will show that the complexity of **P2TCT1** remains an open question. The key lies in the difference between reasonable encoding schemes for problem instances of **P2TCT1** and **P2TCT2**. An instance of **P2TCT1** can be represented as $I_1 = \{S_1, \ldots, S_B; (p_{11}, \ldots, p_{1N_1}), \ldots, (p_{B1}, \ldots, p_{BN_B})\}$. Due to the requirement of identical job processing times, an instance for **P2TCT2** can be more compactly represented as $I_2 = \{S_1, \ldots, S_B; N_1, \ldots, N_B; p\}$. Letting $\bar{S} = \sum_i S_i/B$, $\bar{N} = \sum_i N_i/B$, and $\bar{p} = \sum_i p_{ij}/N$, the size of each instance under reasonable encoding (e.g., using a binary alphabet) is:

$$\begin{aligned} |I_1| &= \sum_i \log S_i + \sum_{ij} \log p_{ij} + N + B \\ &= O(B \log \bar{S} + B\bar{N} \log \bar{p}), \text{ and} \\ |I_2| &= \sum_i \log S_i + \sum_i \log N_i + \log p + 2B \\ &= O(B \log \bar{S} + B \log \bar{N} + \log p). \end{aligned}$$

Notice for an arbitrary instance with identical job processing times, that $|I_1|$ is not bounded from above by a polynomial function of $|I_2|$. For example, if $\bar{S} \leqslant \bar{N}$ and $p = 1$, then $|I_1| = O(B\bar{N}) = O(B2^{\log \bar{N}})$ and $|I_2| = O(B \log \bar{N})$.

We let $\mathbf{P2TCT1_d}$ and $\mathbf{P2TCT2_d}$ denote the decision versions of **P2TCT1** and **P2TCT2**, respectively. Cheng and Chen establish the *NP*-hardness of **P2TCT2** by defining a polynomial time reduction from a known binary *NP*-complete problem (i.e., the Partition Problem) to $\mathbf{P2TCT2_d}$. Their complexity proof implicitly assumes the use of a high-multiplicity encoding scheme for both inputs and outputs; it depends critically on the fact that a reasonable encoding scheme for **P2TCT2** exploits the requirement of identical job processing times, and the reduction ceases to be polynomial if the form illustrated in I_1 is used in place of I_2. The implication of the above is that a polynomial time (in $|I_1|$) algorithm for **P2TCT1** is not necessarily polynomial in $|I_2|$ when applied to solve an instance of **P2TCT2**. Thus, the complexity of **P2TCT1** remains open to further investigation. We note, however, that the question of complexity applies to the case of a fixed number of machines as the problem with an arbitrary number of machines is known to be unary *NP*-hard (Webster 1997).

REFERENCES

Cheng, T. C. E., and Z.-L. Chen. 1994. Parallel Machine Scheduling with Batch Setup Times. *Opns. Res.* **42,** 1171–1174.

Garey, M. R., and D. S. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York.

Webster, S. T. 1997. The Complexity of Scheduling Job Families About a Common Due Date. *Opns. Res. Letters,* **20,** 65–74.

Subject classifications: Analysis of algorithms, computational complexity: *NP*-hardness. Production/scheduling: parallel machine scheduling.
Area of review: Manufacturing.

Operations Research
Vol. 46, No. 3, May–June 1998

0030-364X/98/4603-0423 $05.00