Available online at www.sciencedirect.com
ScienceDirect
European Journal of Operational Research 187 (2008) 1282–1292
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/ejor

# Open-shop batch scheduling with identical jobs

Gur Mosheiov [a,*], Daniel Oron [b]

[a] *School of Business Administration and Department of Statistics, The Hebrew University of Jerusalem, Mount Scopus, Jerusalem 91905, Israel*
[b] *Econometrics and Business Statistics, The University of Sydney, Australia*

Received 11 July 2005; accepted 23 March 2006
Available online 3 November 2006

**Abstract**

This paper addresses batch scheduling problems on an $m$-machine open-shop. The objectives are minimum makespan and minimum flowtime. We assume identical processing time jobs, machine- and sequence-independent setup times and batch availability. The minimum makespan problem is shown to be solved in *constant time*. Specifically, we show that the optimal number of batches is either $m$ or $\left\lceil \frac{n}{\lfloor n/m \rfloor} \right\rceil$ (where $m$ and $n$ are the number of machines and the number of jobs, respectively). The complexity of the minimum flowtime problem is unknown. We propose an $O(n)$ time algorithm which extends the solution of the single-machine case, and produces close-to-optimal solutions.


*Keywords:* Batch scheduling; Open-shop; Makespan; Flowtime

## 1. Introduction

We consider *batch scheduling* problems on an $m$ machine open-shop. The objectives are minimum makespan and minimum flowtime. In batch scheduling a large number of jobs may be grouped and processed as separate batches. Such grouping is generally based on the existence of some similarity between jobs belonging to the same class or family (such as their color or size). A changeover or *setup* time is incurred when starting a new batch. In an open-shop setting, both job allocation to batches and the route of each batch through the different machines are to be determined. There has been an extensive research on batch scheduling and it's various applications, especially in the last two decades; see the recent surveys of Allahverdi et al. [1] and Potts and Kovalyov [9]. However, only a few papers have dealt with open-shop settings: Kleinau [3] studied makespan minimization on a two-machine open-shop with sequence- and machine-independent setup times; Glass et al. [2] studied the two-machine open-shop with sequence-independent and machine-dependent setup times and ''batch availability'' (see below) to minimize makespan. Both problems were shown to be binary NP-hard.

* Corresponding author. Tel.: +972 2 5883790; fax: +972 2 5881341.
*E-mail address:* msomer@mscc.huji.ac.il (G. Mosheiov).

0377-2217/$ - see front matter 
doi:10.1016/j.ejor.2006.03.068

The current paper focuses on the special case of *identical* processing time jobs. This case is highly relevant in many applications, including the common application of repetitive production of the same item. The setup times are assumed to be constant, i.e., *machine*- and *sequence-independent*. (Possible) real life applications for this model are that of a manufacturing or a testing facility, say for circuit boards. Each board must undergo $m$ different tests for quality assurance purposes. There are $m$ different testing stations (or machines), each capable of performing one of the tests. The boards are set on trays, and whenever a tray of boards reaches a station it must be mounted on the machine (requiring a fixed setup time). After the initial setup, the boards are tested one by one, all requiring the same processing time. The assumption that the testing (or processing) time of the boards on all machines is identical (and equal to $p$) can be based on a policy of "equal load" design of such systems.

Very few studies have considered this setting on shops. Makespan minimization with identical jobs and constant setups on an $m$-machine flowshop was shown to be solved in $O(n)$ time, where $n$ is the number of jobs [5]. A faster $O(\sqrt{n})$ algorithm for the flowshop problem was recently designed by Ng and Kovalyov [8]. (Clearly, these algorithms are not polynomial in the size of the input which contains the number of jobs, the number of machines, the identical processing time and the constant setup.) Mosheiov and Oron [5] showed that the 2-machine job-shop problem can be solved in $O(n)$ as well. The case of an $m$-machine job-shop ($m \geqslant 3$) with identical jobs has been shown to be strongly NP-hard even with no batching and setup times [4]. It appears that makespan minimization on an $m$-machine open shop is the only classical machine setting which has not been studied.

Flowtime minimization on a single machine has been studied by several researchers in the last two decades; see Mosheiov et al. [7] and references therein. The flowshop setting has been solved in Mosheiov et al. [6]. To our knowledge, the open-shop case has never been studied.

Our paper considers both problems of minimum makespan and minimum flowtime. For the makespan problem, based on several properties of an optimal schedule, we introduce *a constant time* algorithm. We show that there are only two candidates for the optimal number of batches: for sufficiently large setup time, $m$ (the number of machines) is optimal; otherwise, the optimal number of batches is given by $\left\lceil \frac{n}{\lfloor n/m \rfloor} \right\rceil$. Thus, the makespan as a function of the number of batches attains a global minimum for one of these values. We are not aware of other scheduling problems having similar solution structures.

The complexity status of the flowtime problem is still unknown. We propose an algorithm which extends the solution for the single-machine case. (The latter was introduced first by Santos and Magazine [10] for non-integer batches, and later modified to obtain integer batches by Shallcross [11].) It is clear that our algorithm is very efficient (requiring $O(n)$ time) and produces very close-to-optimal solutions. However a formal analysis of the performance of our algorithm remains a challenging question for future research.

The paper is organized as follows: in Section 2 we introduce the notation and formulation of both problems. In Section 3, we study the makespan minimization problem, and in Section 4, we design our flowtime minimization algorithm.

## 2. Formulation

$n$ jobs, available at time zero, need to be processed on an $m$-machine open-shop. Each job consists of $m$ operations, whose order on the machines is to be determined. All the operations have identical processing times, i.e., $p_{ij} = p$, where $p_{ij}$ denotes the processing time of job $i$ on machine $j$. Preemption is prohibited. Jobs may be grouped into batches whose size is to be determined. A constant integer valued setup time, denoted by $s$, is incurred whenever a batch starts processing on any of the machines. The total processing time of a batch is equal to the sum of processing times of all jobs contained in the batch and the required setup time.

For a given job allocation to batches and a schedule, let $K$ denote the number of batches, and let the pair $(k,j)$ denote batch $k$ processed on machine $j$, $k = 1,\ldots,K$, $j = 1,\ldots,m$. $n_{kj}$ denotes the number of jobs assigned to batch $k$ on machine $j$, i.e., the size of batch $(k,j)$. We assume that a batch remains unchanged on all the machines, i.e. $n_{kj} = n_k$, $j = 1,\ldots,m$ (*batch consistency*). $K$ and $n_k$, $k = 1,\ldots,K$ are decision variables. We assume *batch availability*, i.e., the completion time of a job on a given machine is defined as the completion time of the batch to which it is assigned on that machine. Thus, the (final) completion time of a job is defined as the completion time of the batch to which it is assigned on all the machines. Finally, we assume *non-antic-*