



Total completion time minimization in two-machine flow shop scheduling problems with a fixed job sequence

F.J. Hwang^a, M.Y. Kovalyov^b, B.M.T. Lin^{a,*}

^a Institute of Information Management, Department of Information and Finance Management, National Chiao Tung University, Taiwan

^b United Institute of Informatics Problems, National Academy of Sciences of Belarus, Belarus

ARTICLE INFO

Article history:

Received 9 January 2010

Received in revised form 11 November 2011

Accepted 14 November 2011

Available online 6 December 2011

Keywords:

Two-machine flow shop

Total completion time

Fixed sequence

Dynamic programming

ABSTRACT

This paper addresses scheduling n jobs in a two-machine flow shop to minimize the total completion time, subject to the condition that the jobs are processed in the same given sequence on both machines. A new concept of optimal schedule block is introduced, and polynomial time dynamic programming algorithms employing this concept are derived for two specific problems. In the first problem, the machine-2 processing time of a job is a step increasing function of its waiting time between the machines, and a decision about machine-1 idle time insertion has to be made. This problem is solved in $O(n^2)$ time. In the second problem, the jobs are processed in batches and each batch is preceded by a machine-dependent setup time. An $O(n^5)$ algorithm is developed to find an optimal batching decision.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

This paper studies two scheduling problems in a two-machine flow shop, subject to the condition that the jobs are processed in the same fixed sequence on both machines. A new *block approach* is proposed and deployed to determine their solutions in polynomial time within the general dynamic programming framework.

1.1. State of the art

In machine scheduling, a schedule specifies job starting and completion times on the machines. In some cases, it is completely characterized by the job sequence, and in the other cases, extra information is needed to fully specify it. A complicated schedule structure is typical for two major types of scheduling: scheduling with inserted machine idle time [1] and batch scheduling [2]. A commonly used approach to tackle NP-hard problems of these types is to derive an optimal polynomial time algorithm for the problem in which jobs are processed in the same fixed sequence. Then this algorithm can be used in a local search method to evaluate candidate job sequences for the general problem. There also exist studies, in which a specific optimal job sequence is established analytically. In real-life applications, a job sequence can be given as an input data of the problem. Shafransky and Strusevich [3] studied several open shop scheduling problems, subject to a given sequence of jobs on one machine, and mentioned that a pre-assigned sequence of jobs could be retained on one of the machines in the manufacturing process owing to technological or managerial decisions. A fixed job sequence can also appear due to the application of the First-Come–First-Served principle, which is commonly regarded fair by customers.

* Corresponding author. Tel.: +886 3 5131472; fax: +886 3 5723792.

E-mail addresses: fengjanghwang@gmail.com (F.J. Hwang), kovalyov_my@newman.bas-net.by (M.Y. Kovalyov), bmtlin@mail.nctu.edu.tw, bmtlin@gmail.com (B.M.T. Lin).

Algorithms with machine idle time insertion for a fixed job sequence, also called timing algorithms [4], were applied for a single machine [5–11], parallel machines [12] and a flow shop [13,4]. The problem studied by Lin and Cheng [13] belongs to the area of scheduling with time-dependent processing times, as the machine-2 processing time depends on job's waiting time between the machines. To avoid extra processing on machine 2, idle times can be inserted on machine 1. Scheduling with time-dependent processing times was surveyed by Cheng et al. [14].

Batching decisions for a fixed job sequence were studied by Cheng et al. [15] and Ng and Kovalyov [16] for the makespan minimization in a flow shop. Considering the fabrication and assembly of components in a two-machine flow shop, Cheng and Wang [17] studied the problem of batching the jobs sequenced according to Johnson's rule or the agreeable processing time condition for makespan minimization. As for the assembly-type flowshop batching problem with a fixed job sequence, Lin et al. [18] investigated the makespan minimization for the three-machine case. Later, Hwang and Lin [19] addressed the four regular objective functions of total completion time, maximum lateness, total tardiness, and number of tardy jobs for the general $(m + 1)$ -machine case. Lin and Cheng [20] also considered the objective of maximum lateness, weighted number of tardy jobs or total weighted completion time for a fixed job sequence with centralized or decentralized batching decisions in concurrent open shops.

In this paper, we consider two problems of total completion time minimization in a two-machine flow shop, subject to a fixed job sequence. Though there exist results for the makespan (C_{\max}) minimization in these settings, there is no result for the minimization of total completion time, which is one of the most popular and practically relevant criteria in the scheduling literature.

1.2. Formulation of problems and block approach

Denote the processing times of job j on machine 1 and machine 2 by p_{1j} and p_{2j} , respectively. The first problem includes a step increasing job processing time on machine 2, which depends on the job waiting time between the machines. We call this waiting time a *time lag* between the two operations of a job. For job j , it is denoted by ℓ_j and, given a schedule, it is equal to the difference between the starting time of this job on machine 2 and its completion time on machine 1. A specified time delay t_j is also associated with each job j . If $\ell_j \leq t_j$, then $p_{2j} = \alpha_j$; otherwise $p_{2j} = \alpha_j + \beta_j$. Besides the given job sequence, a schedule should specify machine-1 idle times, which are used to avoid an increase of the job processing times on machine 2, and the time lags between the two operations of each job. Similar to Lin and Cheng [13], who studied the makespan minimization problem in the same setting, our first problem is denoted by $F2|fixed_seq, p_{2j} \in \{\alpha_j, \alpha_j + \beta_j\} | \sum C_j$, where $F2$ stands for a two-machine flow shop, *fixed_seq* for a fixed job sequence, and $\sum C_j$ for the total completion time minimization criterion. The problem is motivated by the manufacturing operations in which temperature regime plays a role.

The second problem is to batch jobs in a predefined sequence. The batches are consistent in the sense that their formations are the same on both machines. Job completion times are determined according to the batch availability model such that jobs of the same batch complete on the same machine at the same time, when processing of the last job in this batch has been finished. The processing time of a batch B on machine l is equal to the sum of the machine setup time, s_l , and job processing times, $\sum_{j \in B} p_{lj}$. The setups are non-anticipatory, i.e., a setup on machine 2 can start only after the corresponding batch is completed on machine 1 and machine 2 is unoccupied. Following Ng and Kovalyov [16], we denote this problem by $F2|fixed_seq, sum_batch, consi, s_l | \sum C_j$, where *sum_batch* stands for the batch processing time calculation formula, *consi* for consistent batches, and s_l for non-anticipatory machine-dependent setup times. Motivation for this problem comes from manufacturing environments in which items move between facilities in containers such as pallets, boxes or carts.

In the two problems under study, the difficulty in the design of a polynomial time dynamic programming algorithm stems from the potential conflict between the makespan and the total completion time. A subschedule which minimizes the total completion time for the first r jobs may retain a relatively large makespan which worsens the total completion time for the remaining $n - r$ jobs. To resolve the difficulty that hinders the principle of optimality, a mechanism for fixing the processing lengths of subschedules is required. A common approach is to introduce temporal parameters into the dynamic programs. This however gives rise to pseudo-polynomial running times. This study develops a two-phase dynamic programming framework without including any temporal parameters as state variables. We present a concept that a feasible schedule can be divided into several specific subschedules, each of which is characterized by its *shape*. On the Gantt chart, the shape of a subschedule in a two-machine flow shop is determined by the positions of its end on machine 1, its start on machine 2 and its end on machine 2, relative to its start on machine 1. Fig. 1 demonstrates that the shape of a subschedule consisting of jobs $\{i, \dots, j\}$ is settled by the three-tuple (L_1, L_2, L_3) . A specific subschedule is called as a *block* and associated with a *state* in the phase-1 dynamic program. A specific state could be associated with more than one block. These blocks in the same state may have different sums of job completion times, while they must retain an identical shape. Based upon this background idea, the concept of *optimal (schedule) block* is introduced. The phase-1 dynamic program produces for each state an *optimal block*, which is a block attaining the minimum total completion time among all blocks associated with the same state. Then an optimal schedule can be built by concatenating several optimal blocks in the phase-2 dynamic program, which is formulated with backward recursion to establish the validity of the principle of optimality in the studied problems.

For the two studied problems, we utilize idle times to divide a feasible schedule into several blocks. Any two consecutive blocks for problems $F2|fixed_seq, p_{2j} \in \{\alpha_j, \alpha_j + \beta_j\} | \sum C_j$ and $F2|fixed_seq, sum_batch, consi, s_l | \sum C_j$ are separated by machine-1 and machine-2 idle times, respectively.

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات