Computers & Operations Research 40 (2013) 2836–2843

Contents lists available at SciVerse ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

A two-stage hybrid flow shop with dedicated machines at the first stage

Jaehwan Yang *

College of Business Administration, University of Seoul, 13 Siripdae-gil, Dongdaemun-gun, Seoul 130-743, Republic of Korea

ARTICLE INFO

Available online 15 June 2013

Keywords:
Two-stage hybrid flow shop
Computational complexity
Heuristic
Worst case bound on the relative error

ABSTRACT

This paper considers the problem of minimizing the makespan in a two-stage hybrid flow shop with dedicated machines at stage 1. There exist multiple machines at stage 1 and one machine at stage 2. Each job must be processed on a specified machine at stage 1 depending on job type, and then the job is processed on the single machine at stage 2.

First, we introduce this problem and establish the complexity of several variations of the problem. For a special case, we show that the decision version is unary NP-complete. For some other special cases, we develop optimal polynomial time solution procedures. Four heuristics based on simple rules such as Johnson's rule and the greedy-type scheduling rule are considered. For each heuristic, we provide some theoretical analysis and find a tight or asymptotically tight worst case bound on the relative error. Finally, the heuristics are empirically evaluated.

1. Introduction

This paper considers the problem of minimizing the makespan in a two-stage hybrid flow shop with dedicated machines at stage 1. There exist multiple machines at stage 1 and one machine at stage 2. Each job must be processed on a specified machine at stage 1 depending on job type, and then the job is processed on the single machine at stage 2. If the job can be processed only by a subset of machines, then the environment is referred to as dedicated machines or machine eligibility (Ribas et al. [12]). In this paper, we say a machine is *dedicated* to a specific job type if such a job can be processed only by the corresponding machine dedicated to that job. No job preemption is allowed and no setup times exist between two stages. All jobs and machines are available at time zero.

Scheduling problems for general hybrid flow shops are different from those for regular flow shops because the former have more than one machine at least at one stage. In real world, scheduling problems for hybrid flow shops are common. An example can be found in some flexible manufacturing systems whose each production stage is either a flexible machine or a flexible manufacturing cell (Lee and Variaktarakis [8] and Zijm and Nelissen [15]). Another example can be found in the process industry where multiple servers (machines) are available at each stage (Brah and Hunsucker [1]). For an in-depth review of the industries where hybrid flow shops are prevalent and more extensive literature review, see Ribas et al. [12] and Ruiz and Vazquez-Rodriguez [13].

The problem in this paper is different from all other hybrid flow shop problems because there exist dedicated machines at stage 1. The presence of dedicated machines at stage 1 is common in the real world. For example, various types of products are processed on different machines during the manufacturing stage depending on product specifications and then the products go through the same testing and inspection process (Oğuz et al. [11]) or a packaging process at the end of the final assembly. Another example can be found at restaurant kitchens. Various food items are prepared using different cooking tools but at the final step of the cooking process, the food items may have to go through the same oven or burner.

Several studies have considered this particular type of the problem with the objective of minimizing the makespan. Oğuz et al. [11] consider the problem where there exist two dedicated machines that are identical at stage 1. They show that the problem is at least binary NP-complete, and develop a heuristic called H1. For the H1, they prove that the asymptotically tight bound on the relative error is 3/2. Later, Lin [9] establishes that the decision version of the problem is unary NP-complete even if there exist only two dedicated machines that are identical at stage 1. Hsu et al. [6] consider the case where there exist multiple identical dedicated machines at stage 1. For the case where there exist arbitrary number of machines at stage 1, they show that the problem is unary NP-complete without recognizing the complexity result in Lin [9]. They develop several heuristics based on simple scheduling rules and evaluate them empirically. Low et al. [10] consider the case where there exist multiple number of

* Tel.: +82 2 6490 2236; fax: +82 2 2246 0570.
E-mail address: jyang@uos.ac.kr

0305-0548/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cor.2013.05.020

unrelated dedicated machines at stage 1. Their analysis and results are similar to those in Hsu et al. [6]. Finally, He and Sun [5] consider the scheduling of semiconductor burn-in operations where there exist multiple dedicated machines that are identical at stage 1 and a batch processor at stage 2. The burn-in oven is modeled as a batch-processing machine, and the processing time of a batch is defined as the longest processing time for jobs belonging to that batch. They develop a procedure for an optimal polynomial time solution for some special cases, and also develop and analyze a heuristic.

This paper considers the hybrid flow shop scheduling problem where there exist multiple dedicated machines that are identical at stage 1 and one machine at stage 2. We consider some special cases based on the characteristics of the processing time at each stage. For one such case, we show that the decision version is unary NP-complete. For some other special cases, we develop polynomial time optimal solution procedures. Four heuristics based on simple rules such as the Johnson's rule and the greedy-type scheduling rule are developed. For each heuristic, we provide some theoretical analysis and find either a tight or asymptotically tight worst case bound on the relative error. In particular, heuristic H1, which is for the case with two dedicated machines at stage 1 by Oğuz et al. [11], is extended to the case where there exist multiple machines at stage 1 and an asymptotically tight worst case bound on the relative error is found. Then, we empirically evaluate the heuristics. Note that some of the analytical techniques used in this paper are similar to those in Yang [14], who considers a scheduling problem with the objective of minimizing the total completion time in the hybrid flow shop with a single machine at stage 1 and two dedicated machines that are identical at stage 2.

The rest of this paper is organized as follows. Section 2 introduces the notations, and Section 3 presents the preliminary results. In Section 4, we establish the complexity of one special case and find polynomial time optimal solution procedures for several other special cases. In Sections 5 and 6, we develop several lower bounds for the problem and three intuitive heuristics for the general case, respectively. Then, for each heuristic of the heuristics and an algorithm from a previous section, we analyze it and find a worst case bound on the relative error. Finally, we empirically evaluate the heuristics, provide a summary, and discuss some interesting avenue for future research.

2. Notation

The decision variables in our model are

σ_i	schedule of all jobs on machine i for $i \in \{1, 2, \ldots, m+1\}$ where m is the number of machines at stage 1
σ	schedule of all jobs.

Other notation includes

n	number of jobs
m	number of machines at stage 1
N	set of jobs $= \{1, 2, \ldots, n\}$
M	set of machines at stage $1 = \{1, 2, \ldots, m\}$
M_i	machine i for $i \in M \cup \{m+1\}$
n_i	number of jobs processed on M_i and M_{m+1} for $i \in M$
N_i	set of jobs processed on M_i and M_{m+1} for $i \in M$
p_{ij}	processing time of job j on M_i for $i \in M \cup \{m+1\}$ and $j \in N$
$C_j(\sigma_i)$	completion time of job j on machine i in schedule σ for $i \in M \cup \{m+1\}$ and $j \in N$
$C_j(\sigma)$	completion time of job j in schedule σ for $j \in N$
σ^*	an optimal schedule
z^*	value of optimal schedule σ^*

Observe that p_{ij} does not exist if job j does not belong to N_i for $i \in M$ and $j \in N$, and $n = n_1 + n_2 + \cdots + n_m$. Also, jobs in N_i can be processed only by M_i for $i \in M$ at stage 1. When there exists no confusion, we replace $C_j(\sigma)$ and $C_j(\sigma_i)$ with C_j and C_{ij}, respectively. Here $[j]$ indicates the job in the jth position in schedule σ. For example, $p_{1[4]}$ is the processing time on M_1 of the fourth job in schedule σ. We classify our problem according to the standard classification scheme for scheduling problems (Graham et al. [3]). In the three field notation of $\alpha_1|\alpha_2|\alpha_3$, α_1 describes the machine structure, α_2 gives the job characteristics and restrictions, and α_3 defines the objective. This scheme is extended to the scheduling problem for a two-stage hybrid flow shop scheduling in the α_1 field as suggested by Gupta et al. [4]. Following the standard scheduling classification scheme of Graham et al. [3] and the suggestion by Gupta et al. [4], we refer to the problem of minimizing the makespan in a two-stage hybrid flow shop with arbitrary number of machines at stage 1 and one machine at stage 2 as $F2(P,\ 1)||C_{max}$. Because our problem entails dedicated machines at stage 1, we denote the problem as $F2(P,1)||C_{max}$ with dedicated machines.

A *schedule* defines a job order for each machine and a *permutation schedule* is a schedule in which stages 1 and 2 have the same processing order. For $F2(P,1)||C_{max}$ with dedicated machines, the jobs are available at the start of the planning process. In addition, no preemptions are allowed.

3. Preliminary results

This section establishes some properties of an optimal schedule. The *inserted idle time* occurs when a machine is intentionally kept idle even if there exists a waiting job. Because there are no restrictions that delay jobs, we have the following result.

Lemma 1. *For problem $F2(P,1)||C_{max}$ with dedicated machines, there exists an optimal schedule without inserted idle time on M_i for $i \in M \cup \{m+1\}$.*

The following lemma establishes that there exists an optimal permutation schedule.

Lemma 2. *For problem $F2(P,1)||C_{max}$ with dedicated machines, there exists an optimal permutation schedule.*

Proof. A simple pairwise interchange argument proves the result. □

As a result of Lemma 2, we only consider a permutation schedule. Then, a schedule can be fully described by the job order.

The heuristic procedures that are developed employ the well-known rule called Johnson's rule (Johnson [7]), which is optimal for problem $F2||C_{max}$. This rule can be briefly described as follows.

Johnson's rule

1. List jobs and their processing times on each machine.
2. If the shortest processing time for either machine is for the first machine, then schedule the corresponding job first. Otherwise, schedule the corresponding job last. Break ties arbitrarily.
3. Remove the scheduled job from the list, and repeat Step 2 until no unscheduled jobs are left.

4. Complexity and special cases

The general version of the problem is unary NP-complete (Lin [9]), and therefore this section establishes the complexity of several special cases. Recall that the Johnson's rule generates an