Computers & Operations Research 39 (2012) 1605–1614

Contents lists available at SciVerse ScienceDirect

# Computers & Operations Research

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

# Mathematical models for preemptive shop scheduling problems

Abbas Ebadi, Ghasem Moslehi *

*Department of Industrial and Systems Engineering, Isfahan University of Technology, Isfahan 84156-83111, Iran*

ARTICLE INFO

Available online 29 September 2011

*Keywords:*
Job shop scheduling problem
Flow shop scheduling problem
Preemption
Integer Linear Programming

ABSTRACT

More than half a century has passed since Bowman and Dantzig (1959) [13,14] introduced their models for preemptive shop scheduling problems. A more efficient model seems to be needed to address all the aspects involved in the problem. We introduce a new Integer Linear Programming (ILP) formulation as a new method for solving the preemptive Job Shop Scheduling Problem (pJSSP). The dimension of the new model, unlike those of the existing ones, depends solely on the number of jobs and machines irrespective of processing times. The proposed model is used as an optimal, two-phase approach. In phase one, the model is solved to obtain the start and completion times of each operation on each machine. In phase two, a simple algorithm in $O(mn \log n)$ steps is used to turn these times into a complete and optimal schedule. Different preemptive flow shop problems are studied as special cases of the pJSSP while some related properties are also discussed. Finally, the higher efficiency of the proposed model is verified both theoretically and computationally through its comparison with conventional methods commonly in use.



## 1. Introduction

In job shop environments, system efficiency is influenced to a great extent by scheduling since a wide variety of products are to be produced and a specific operational route is to be assigned to each product. On the other hand, one of the most important assumptions in scheduling is preemption that can be used to improve the objective function. Preemption is allowed in machining operations where the amount of processing a job receives is not lost after preemption. In addition to its manufacturing applications, pJSSP has found applications in such project management areas as allocating of human resources to different project activities. Guo et al. [1] used a special case of this problem in the apparel industry and Anderson et al. [2] used the 2-machine version of the problem to schedule computer systems. Preemption may also decrease computational complexity in some problems, but not in the Job Shop Scheduling Problem (JSSP). For example, the two machine pJSSP with only three jobs ($J2/n=3, prmp/C_{max}$) is an NP-hard problem but the non-preemptive version of this problem with any arbitrary number of jobs ($J2/n=k/C_{max}$) is solvable in a polynomial time [3]. The non-preemptive JSSP becomes an NP-hard problem when there are at least three jobs and three machines ($J3/n=3/C_{max}$) [4].

Mathematical programming is an exact method used to solve scheduling problems. By this method, a basic framework is provided for a wide variety of problems and different objective functions and constraints are then introduced into the model. Furthermore, in a mathematical model aimed at minimizing the objective function, the feasible region increases if certain constraints are relaxed so that the value of the objective function may decrease and a lower bound may be obtained for the problem. Feasible solutions obtained by other methods can also be sometimes used as a starting point for solving the mathematical models. Integer Programming (IP) models have become increasingly important in recent years thanks to the development of efficient algorithms, robust software systems, and advanced computers. The deadlock in the field was probably due to the absence of these developments in the past. For instance, the IP model for the non-preemptive JSSP proposed by Manne [5] was recently improved by Liao and You [6] after 30 years, which encouraged more researchers to develop models for non-preemptive problems. Pan and Chen [7] developed a Mixed Binary Integer Programming (MBIP) model for the reentrant JSSP based on the models developed by Manne [5] and Liao and You [6]. Dessouky and Leachman [8] developed dynamic models based on previous models for a problem with more than one machine of each type but with identical products. Fattahi et al. [9] developed a mathematical model and a heuristic approach for the flexible JSSP (FJSSP). Gomez et al. [10] proposed an IP model for the FJSSP in which certain assumptions including limited intermediate buffers, similar parallel machines, recirculation, and flexible processing routes were considered. They used the commercial MILP software

* Corresponding author. Tel.: +98 311 3915509; fax: +98 311 3915526.
*E-mail addresses:* a.ebadi@in.iut.ac.ir (A. Ebadi), moslehi@istt.ir, moslehi@cc.iut.ac.ir (G. Moslehi).

0305-0548/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cor.2011.09.013

for solving the model. Ding et al. [11] and Rong-Hwa [12] developed a meta-heuristic and mathematical model by considering dissimilar parallel machines with internal due dates to minimize earliness and tardiness.

Mathematical models for preemptive job shop scheduling problems have received far lower attention than the non-preemptive ones to the extent that, now after 50 years, the models introduced by Bowman [13] and Dantzig [14] not only still retain their original forms but have also been pushed into neglect. This may be due to their low efficiency and their failure to solve even small sized problems. The dimension of these models depends not only on the number of jobs and machines but also on the scheduling horizon ($T$). $T$ is an upper bound for the makespan which is, in turn, a function of processing times. Hence, the dimension of the existing models depends on processing times. The value of $T$ should be determined at the beginning of the solution process; for example, it can be assumed to be equal to the total processing time. Hence, the restriction on $T$ makes it impossible to solve problems of even the smallest size with large processing times. A new mathematical model is, therefore, required whose dimension does not depend on processing times. It is the objective of this paper to develop one such model. Throughout this paper, the scheduling problem is designated by the triple notation $\alpha/\beta/\gamma$ [15].

The paper is organized as follows. In Section 2, the proof for the theorems that form the basis of our two-phase approach will be presented. In Section 3, a primitive NonLinear Programming (NLP) model will be developed for the problem $J/prmp/C_{max}$, and in Section 4, the model will be converted into a final MBIP one by introducing certain binary variables and constraints. In Section 5, the flow shop problems will be studied as special cases of job shop problems for which our proposed model is especially tailored. In Section 6, the dimensions of the proposed model will be compared with those of the Bowman and Dantzig's. In Section 7, the new model will be employed for solving different problem sizes and the results obtained will be compared with those from the best model available. Finally, in Section 8, conclusions will be presented and some suggestions will be made for future studies.

## 2. Problem definition and properties

In a pJSSP, there are $n$ jobs and $m$ machines. Each job has its own sequence of operations, and each operation should be processed on a particular machine. The objective is to schedule operations on machines so that maximum completion time is minimized. Problem assumptions can be stated as follows. Processing times are deterministic and sequence-independent. All jobs are ready to be processed at time zero. Only one job can be processed on each machine at a given time. Each job visits each machine once at most and preemption is allowed, i.e. processing of any operation may be interrupted to be resumed later. First, consider the following notations:

| | |
|---|---|
| $n$ | number of jobs |
| $m$ | number of machines |
| $J_i$ | job number $i$ ($i=1,2,\ldots,n$) |
| $J$ | set of all jobs ($J=\{J_i \mid i=1,2,\ldots,n\}$) |
| $M_k$ | machine number $k$ |
| $p_{i\ k}$ | processing time of $J_i$ on $M_k$ ($i=1,2,\ldots,n$ , $k=1,2,\ldots,m$) |
| $q_{i\ l\ k}$ | 1 if the $l$th operation of $J_i$ requires $M_k$; 0, otherwise ($i=1,2,\ldots,n$ , $l=1,2,\ldots,m$ , $k=1,2,\ldots,m$) |
| $A$ | an arbitrary schedule for the problem $j/prmp/C_{max}$ |
| $A_k$ | schedule on $M_k$ in $A$, so $A=\{A_1,A_2,\ldots,A_m\}$ |
| $S_{ik}^A$ | start time of $J_i$ on $M_k$ in $A_k$ ($i=1,2,\ldots,n$ , $k=1,2,\ldots,m$) |
| $C_{ik}^A$ | completion time of $J_i$ on $M_k$ in $A_k$ ($i=1,2,\ldots,n$ , $k=1,2,\ldots,m$) |
| $R_{ik}^A$ | ready time of $J_i$ on $M_k$ which is determined according to schedule $A_k$; $R_{ik}^A=S_{ik}^A$ |

"Ready time" is the time when a job becomes ready to be processed.

| | |
|---|---|
| $D_{ik}^A$ | due date of $J_i$ on $M_k$ determined according to schedule $A_k$; $D_{ik}^A=C_{ik}^A$. |
| $T_{max}$ | maximum tardiness |
| $B$ | a schedule for the problem $J/prmp/C_{max}$, where the schedule on any machine $k(k=1,2,\ldots,m)$ is obtained by using times $R_{ik}^A$ and $D_{ik}^A$, and optimally solving the problem $1/R_{ik}^A,prmp/T_{max}$. |
| $B_k$ | schedule on $M_k$ in $B$, so $B=\{B_1,B_2,\ldots,B_m\}$ |
| $C_{max}^A$ | maximum completion time or makespan of schedule $A$; $C_{max}^A=\max_{1\le i\le n,\ 1\le k\le m} C_{ik}^A$ |
| $C_{max}^B$ | maximum completion time or makespan of schedule $B$; $C_{max}^B=\max_{1\le i\le n,\ 1\le k\le m} C_{ik}^B$ |
| $T_{max}^{B_k}$ | maximum tardiness in schedule $B_k$; $\max_{1\le i\le n}(\max\{0,C_{ik}^B-D_{ik}^B\})$ |

According to the definition, only times $S_{ik}^A$ and $C_{ik}^A$ are used in deriving schedule $B$ from schedule $A$; preemption times in schedule $A$ have, therefore, no influence on schedule $B$. In the following theorems, it is proved that $B$ is a feasible schedule which observes the condition $C_{max}^B\le C_{max}^A$.

**Theorem 1.** *For any machine, $k$, if $S_{ik}^A$ and $C_{ik}^A$ are regarded as $R_{ik}^A$ and $D_{ik}^A$, respectively, and the problem $1/R_{ik}^A,prmp/T_{max}$ is solved optimally, then Schedule $B_k$ obtains with $T_{max}^{B_k}=0$.*

**Proof.** If a machine $k$ exists such that solving the problem $1/R_{ik}^A,prmp/T_{max}$ for it optimally yields $T_{max}>0$, it can be inferred that there is at least one tardy job in any schedule on machine $k$. However, this result is impossible due to Schedule $A_k$ in which $C_{ik}^A=D_{ik}^A$ for every $i$. Therefore, when the problem $1/R_{ik}^A,prmp/T_{max}$ is optimally solved on machine $k$, the jobs are scheduled without any tardiness. □

Now, it can be shown that schedules $B_k$, i.e. $\{B_1,B_2,\ldots,B_m\}$, which were separately obtained on different machines, constitute a feasible schedule for the problem $J/prmp/C_{max}$. In other words, $B=\{B_1,B_2,\ldots,B_m\}$ is a feasible schedule. Furthermore, it can be shown that $C_{max}^B\le C_{max}^A$ is valid.

**Theorem 2.** *If schedules, $B_k(k=1,2,\ldots,m)$, are constructed according to Theorem 1, then the feasible schedule $B$ with $C_{max}^B\le C_{max}^A$ obtains for the problem.*

**Proof.** Consider the following:

- In constructing schedules $B_k$, $S_{ik}^A$ was regarded as $R_{ik}^A$. So, $S_{ik}^B\ge S_{ik}^A$ is true for every $i$ in schedules $B_k$.
- In constructing schedules $B_k$, $C_{ik}^A$ was regarded as $D_{ik}^A$, and according to Theorem 1, there is no tardy job in $B_k$. Therefore, $C_{ik}^B\le C_{ik}^A$ is true for every $i$ in schedules $B_k$.

It is, therefore, concluded that the jobs in Schedule $B$ are scheduled between $S_{ik}^A$ and $C_{ik}^A$ on any machine $k$. According to definition, $A$ is a feasible schedule with $C_{max}^A$. Thus, Schedule $B$ is a feasible schedule whose makespan is not greater than $C_{max}^A$. □

**Remark 1.** The optimal schedule for the problem $1/R_{ik}^A,prmp/T_{max}$ is attainable using the preemptive Earliest Due Date (pEDD) rule [15].