Computers & Operations Research 41 (2014) 94–97

Contents lists available at ScienceDirect

Computers & Operations Research

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

Block models for scheduling jobs on two parallel machines with a single server

Keramat Hasani [a], Svetlana A. Kravchenko [b], Frank Werner [c,*]

[a] Islamic Azad University, Malayer Branch, Malayer, Iran
[b] United Institute of Informatics Problems, Surganova St. 6, 220012 Minsk, Belarus
[c] Fakultät für Mathematik, Otto-von-Guericke-Universität Magdeburg, Postfach 4120, 39016 Magdeburg, Germany

ARTICLE INFO

Available online 20 August 2013

Keywords:
Scheduling
Parallel machines
Single server

ABSTRACT

We consider the problem of scheduling a set of non-preemptable jobs on two identical parallel machines such that the makespan is minimized. Before processing, each job must be loaded on a machine, which takes a given setup time. All these setups have to be done by a single server which can handle at most one job at a time. For this problem, we propose a mixed integer linear programming formulation based on the idea of decomposing a schedule into a set of blocks. We compare the results obtained by the model suggested with known heuristics from the literature.

1. Introduction

The problem considered in this paper can be described as follows. There are n independent jobs and two identical parallel machines. For each job J_j, $j=1,\ldots,n$, there is given its processing time p_j. Before processing, a job must be loaded on the machine M_q, $q=1,2$, where it is processed which requires a setup time s_j. During such a setup, the machine M_q is also involved into this process for s_j time units, i.e., no other job can be processed on this machine during this setup. All setups have to be done by a single server which can handle at most one job at a time. The goal is to determine a feasible schedule which minimizes the makespan. So, using the common notation, we consider the problem $P2,S1\|C_{max}$. This problem is strongly NP-hard since problem $P2,S1|s_j=s|C_{max}$ is unary NP-hard [5]. The interested reader is referred to [3,6] for additional information on server scheduling models.

Several heuristics were developed for the problem $P2,S1\|C_{max}$ under consideration so far. In Abdekhodaee et al. [2], two versions of a greedy heuristic, a genetic algorithm and a version of the Gilmore–Gomory algorithm were proposed and tested. The analysis started in [2] was extended in Gan et al. [4], where two mixed integer linear programming formulations and two variants of a branch-and-price scheme were developed. Computational experiments have shown that for small instances with $n\in\{8,\ 20\}$, one of the mixed integer linear programming formulations was the best whereas for the larger instances with $n\in\{50,\ 100\}$, the branch-and-price scheme worked better, see [4].

In this paper, we propose a mixed integer linear programming formulation for the problem $P2,S1\|C_{max}$, based on the structure of an optimal schedule. The proposed models use the simple idea of a possible decomposition of any schedule into a set of blocks, which significantly contributes to a reduction of the number of jobs. We compare the performance of this model with the heuristics proposed in [4].

The remainder of the paper is organized as follows. In Section 2, we introduce two block models for the problem under consideration and give the resulting mixed integer programming formulations. In Section 3, we present computational results and perform a comparison with existing heuristics. Finally, we give some concluding remarks in Section 4.

2. Block models

It is easy to see that any schedule for the problem $P2,S1\|C_{max}$ can be considered as a unit of blocks $B_1,\ldots,B_z$, where $z\le n$. Each block B_k can be completely defined by the first level job J_a and a set of second level jobs $\{J_{a1},\ldots,J_{ak}\}$, where inequality $p_a\ge s_{a1}+\cdots+s_{ak}+p_{a1}+\cdots+p_{ak}$ holds, see Fig. 1.

For example, for the schedule given in Fig. 2, we can define the four blocks B_1, B_2, B_3, and B_4. The block B_1 is defined by the first level job J_1 and by the second level job $\{J_2\}$; the block B_2 is defined by the first level job J_3 and by the second level job $\{J_4\}$; the block B_3 is defined by the first level job J_5 and an empty set of second level jobs; the block B_4 is defined by the first level job J_6 and by the second level jobs $\{J_7, J_8\}$.

Thus, the model that we suggest is based on the fact that any schedule can be decomposed into a set of blocks. The variable $B_{k,f,j}$

* Corresponding author. Tel.: +49 391 6712025; fax: +49 391 6711171.
E-mail address: frank.werner@mathematik.uni-magdeburg.de (F. Werner).

0305-0548/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cor.2013.08.015

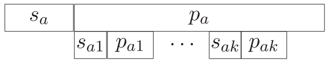

Fig. 1. Each block can be completely defined by the first level job J_a and a set of the second level jobs $\{J_{a1}, \ldots, J_{ak}\}$.

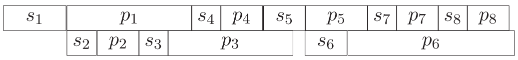

Fig. 2. A schedule with four blocks.

is used for a block. We have $B_{k,f,j}=1$ if job J_j is scheduled in level f in the k-th block, otherwise $B_{k,f,j}=0$. The index $k=1,\ldots,n$ indicates the serial number of the block. The index $f\in\{1,2\}$ indicates the level, i.e., we have $f=1$ if the level is the first one, and $f=2$ if the level is the second one. The index $j=1,\ldots,n$ indicates the job.

Each job belongs to some block, i.e., for any $j=1,\ldots,n$, the equality

$$\sum_{k=1}^{n}\sum_{y=1}^{2}B_{k,y,j}=1 \tag{2.1}$$

holds. There is only one job of the first level for each block, i.e., for each $y=1$ and for any $k=1,\ldots,n$, the inequality

$$\sum_{j=1}^{n}B_{k,1,j}\le 1 \tag{2.2}$$

holds.

Since all blocks are given, we define the following data for each block B_k, where $k=1,\ldots,n$:

- The loading part of the block B_k has the length $ST_k\ge 0$, formally inequality

  $$ST_k\ge\sum_{j=1}^{n}s_jB_{k,1,j} \tag{2.3}$$

  holds.
- The objective part of the block B_k has the length $\sum_{j=1}^{n}(s_j+p_j)B_{k,2,j}$.
- The processing part of the block B_k has the length $PT_k\ge 0$, formally inequality

  $$PT_k\ge\sum_{j=1}^{n}p_jB_{k,1,j}-\sum_{j=1}^{n}(s_j+p_j)B_{k,2,j} \tag{2.4}$$

  holds.

Thus, each block is composed of three parts: *loading*, *objective*, and *processing*.

We add the objective part to the objective function and delete it from the block. After deleting the objective part from each block, the schedule can be considered as a set of modified jobs $J_k^{'}$ with the setup time ST_k and the processing time PT_k. The jobs $J_k^{'}$, $k=1,\ldots,n$, are scheduled in staggered order, i.e., job $J_1^{'}$ is scheduled on the first machine, job $J_2^{'}$ is scheduled on the second machine, job $J_3^{'}$ is scheduled on the first machine, job $J_4^{'}$ is scheduled on the second machine, and so on.

In Fig. 2, we have a schedule consisting of four blocks.

- For the first block we have $B_{1,1,1}=1$, i.e., J_1 is the first level job, and $B_{1,2,2}=1$, i.e., J_2 is the second level job. The modified job $J_1^{'}$ has the loading part $ST_1=s_1$ and the processing part $PT_1=p_1-s_2-p_2$.
- For the second block we have $B_{2,1,3}=1$, i.e., J_3 is the first level job, and $B_{2,2,4}=1$, i.e., J_4 is the second level job. The modified job $J_2^{'}$ has the loading part $ST_2=s_3$ and the processing part $PT_2=p_3-s_4-p_4$.
- For the third block we have $B_{3,1,5}=1$, i.e., J_5 is the first level job, and there are no second level jobs in this block. The modified job $J_3^{'}$ has the loading part $ST_3=s_5$ and the processing part $PT_3=p_5$.
- For the fourth block we have $B_{4,1,6}=1$, i.e., J_6 is the first level job, and there are two jobs J_7 and J_8 of the second level, i.e., $B_{4,2,7}=1$ and $B_{4,2,8}=1$. The modified job $J_4^{'}$ has the loading part $ST_4=s_6$ and the processing part $PT_4=p_6-s_7-p_7-s_8-p_8$.

The jobs $J_1^{'}, J_2^{'}, J_3^{'}, J_4^{'}$ are processed alternately on the two machines.

Formally, if we denote by st_j the starting time of each modified job $J_j^{'}$, then $st_1+ST_1\le st_2$, $st_2+ST_2\le st_3$, and so on, i.e., inequality

$$st_j+ST_j\le st_{j+1} \tag{2.5}$$

holds for each $j=1,\ldots,n-1$;

$st_1+ST_1+PT_1\le st_3$, $st_2+ST_2+PT_2\le st_4$, and so on, i.e., inequality

$$st_j+ST_j+PT_j\le st_{j+2} \tag{2.6}$$

holds for each $j=1,\ldots,n-2$.

We denote by F the total length of the modified schedule, i.e., inequality

$$F\ge st_n+ST_n+PT_n \tag{2.7}$$

holds, and inequality

$$F\ge st_{n-1}+ST_{n-1}+PT_{n-1} \tag{2.8}$$

holds.

For each job J_j, the integer number $ch[j]$ is introduced with the following meaning. If J_j is the first level job for some block B_x, then $ch[j]$ denotes the maximal number of second level jobs for the same block. Formally, one can write

$$B_{x,2,1}+\cdots+B_{x,2,n}\le ch[1]B_{x,1,1}+\cdots+ch[n]B_{x,1,n} \tag{2.9}$$

The objective function is

$$F+\sum_{x=1}^{n}\sum_{j=1}^{n}(s_j+p_j)B_{x,2,j}. \tag{2.10}$$

Since any schedule can be decomposed into a set of blocks, the following theorem holds.

Theorem 1. *Any schedule s can be described as a feasible solution of system (2.1)–(2.8) and as a feasible solution of system (2.1)–(2.9), respectively. In both cases, equality*

$$C_{max}(s)=F+\sum_{x=1}^{n}\sum_{j=1}^{n}(s_j+p_j)B_{x,2,j}$$

holds.

Now, to prove the equivalence between the scheduling problem $P2,S1\|C_{max}$ and the models (2.1)–(2.8) and (2.1)–(2.9), respectively, one has to prove the following theorem.

Theorem 2. *Any feasible solution of system (2.1)–(2.8) and any feasible solution of system (2.1)–(2.9), respectively, can be described as a feasible schedule for the problem $P2,S1\|C_{max}$. In both cases, equality*

$$F+\sum_{x=1}^{n}\sum_{j=1}^{n}(s_j+p_j)B_{x,2,j}=C_{max}(s)$$

holds.

Proof. Suppose that we have some feasible solution of system (2.1)–(2.8). Using the values st_j, ST_j and PT_j, one can reconstruct the schedule s' for the set of modified jobs $J_1^{'}, J_2^{'}, \ldots, J_n^{'}$. Since all these jobs are scheduled in staggered order, it is sufficient to consider only the following three cases for the possible scheduling of two adjacent jobs, say $J_j^{'}$ and $J_{j+1}^{'}$.