Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-12 (2016) 414–418

The chain-reentrant shop with the no-wait constraint

Karim Amrouche [*,**] **Mourad Boudhar** [**] **Farouk Yalaoui** [***]

[*] *University of Economics and Management Sciences, 02 street Ahmed Ouked, Dely Brahim, Algiers, Algeria (e-mail: amrouche-karim@hotmail.com).*
[**] *RECITS Laboratory, Faculty of Mathematics, USTHB University, BP 32 Bab-Ezzouar, El-Alia 16111, Algiers, Algeria, (e-mail: mboudhar@yahoo.fr)*
[***] *University of Technology of Troyes, LOSI laboratory, ICD UMR CNRS 6281, 12 Rue Marie Curie, CS 42060, Troyes 10004, France, (e-mail: farouk.yalaoui@utt.fr)*

Abstract: We consider a chain reentrant shop problem with m machines, in which a no-wait constraint is imposed. Each job goes through m machines in a fixed order and return back to the first machine for its last operation. We seek to minimize the overall finish time (makespan). We first proof some new NP-hard problems in the case of two machines. For the resolution of the general problem we propose a genetic algorithm with numerical experiments.

Keywords: Scheduling, flowshop, chain-reentrant, no-wait, NP-hardness, genetic algorithms.

1. INTRODUCTION

In a typical m-machine flowshop, tasks are composed of m operations and each task visits each machine only once. On the other hand, in re-entrant flowshops, there are re-entrant flows, which means that, tasks should visit machines multiple times. In this paper we shall focus on a chain reentrant shop with the no-wait constraint. Where n independent tasks should be scheduled on a set of m machines, and each task goes for its processing on the first machine called also the primary machine (M_1), then to a number of other machines in a fixed order, and finally return back to the primary machine for its last operation. Note that where the no-wait constraint is imposed that means that tasks should processed continuously from their start on M_1, to their completion on M_1 without any interruption on or between machines. We can meet this type of problems in hot metal rolling, where interruption would preclude the maintenance of continuously high operating temperatures (Rock, 1984).
This problem is denoted: $Fm|chain-reentrant, no-wait|C_{max}$ and the processing pattern of the tasks is given in Fig. 1.
For scheduling tasks denote:

- $a_{[j]}$: The first operation of the task T_j on the first machine.
- $b_{[ji]}$: The operation of the task T_j on the machine M_i $i = 2, \dots m$.
- $c_{[j]}$: The second operation of the task T_j on the first machine.
- a_j: the processing time of the first operation of the task T_j on the first machine.
- b_{ji}: the processing time of the task T_j on the machine M_i $i = 2, \dots m$.
- c_j: the processing time of the second operation of the task T_j on the first machine.

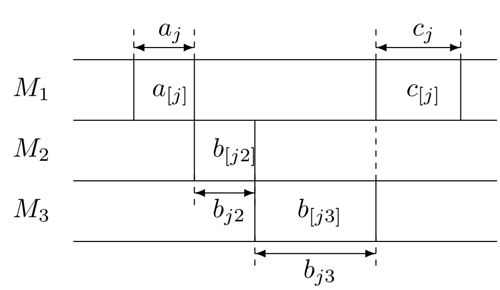

Fig. 1. Processing of the task T_j on machines.

The two machines flow shop with the no-wait constraint is polynomial (Gilmore and Gomory, 1964) and the latter becomes NP-hard in the strong sense where the number of machines is equal to three (Rock, 1984). Sahni and Cho (1979) have proved that the no-wait job shop and the no-wait open shop problem are strongly NP-hard, even if there are only two stages and if each job consists of only two operations. The no-wait permutation flow shop problem can be formulated as an asymmetric traveling salesman problem (ATSP); see e.g. Piehler (1960), Wismer (1972), Hall and Sriskandarajah (1996) have given a detailed survey of the research and applications of no-wait flow shop scheduling problem. Wang et al. (1997) have considered the chain reentrant shop problem without the no-wait constraint, they have proved some properties that identify a specific class of optimal schedules. They have also elaborated an approximation and a branch and bound algorithms.
Amrouche and Boudhar (2016) and Amrouche et al. (2016)

2405-8963 © 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.07.641

have already considered the chain-reentrant flow shop with two machines and exact time lags: $F2|ChR, l_j|C_{max}$. The authors have proved some NP-hardness results and have presented several well solvable cases.

The remainder of this paper is organized as follows. In Section 2, we show that the two machine chain reentrant with the no-wait constraint is NP-hard. In section 3, a metaheuristic algorithm with empirical experiments are provided for the m-machine case. The conclusion constitutes the last section.

2. NP-HARDNESS

In this section we focus our study on the case where the number of machines is equal to two ($m = 2$). Three new ordinary NP-hard problems are given.

In order to prove the NP-hardness of the following problems, we use a reduction from the partition problem known to be ordinary NP-complete (Garey and Johnson, 1979).

Partition
INSTANCE: An integer L, and n positive integers $v_j : j \in V = \{1, 2 \ldots, n\}$ such that
$\sum_{j \in V} v_j = 2L$.
QUESTION: Is there subset $V' \subset V$ such that $\sum_{j \in V'} v_j = \sum_{j \in V \setminus V'} v_j = L$?

Theorem 1. The problem :$F2|chain - reentrant, no - wait, b_j \in \{0, L\}, c_j \in \{0, L+1\}|C_{max}$ is NP-hard.

Proof.

Given an arbitrary instance of the partition problem, we consider the following instance of our scheduling problem, where the number of tasks is equal to $n + 2$ and the processing times are given in Table 1.
Is there exist a schedule σ of the $n + 2$ tasks on the two machines M_1, M_2 with $C_{max}(\sigma) = 6L + 4$?

Table 1. processing times of the tasks

$n+2 : tasks$	a_j	b_j	c_j
$T_j : j = 1, \ldots, n$	$a_j = v_j$	0	0
T_{n+1}	$L+1$	L	$L+1$
T_{n+2}	$L+1$	L	$L+1$

Our problem belongs to NP because we can verify in polynomial time if a permutation of tasks satisfies all the constraints of the problem.
We prove that the scheduling problem has a solution if and only if the partition problem has a solution.
If the partition problem has a solution, then there exists a partition of V into two disjoint subsets $V', V \setminus V'$ such that $\sum_{j \in V'} v_j = \sum_{j \in V \setminus V'} v_j = L$. Suppose without loss of generality that $V' = \{v_1, \ldots, v_k\}$ and $V \setminus V' = \{v_{k+1}, \ldots, v_n\}$.
For this solution, we construct a solution of our problem as shown in Figure 2.

The value of the makespan is equal to $6L + 4$.
Conversely, suppose there is a schedule σ with makespan

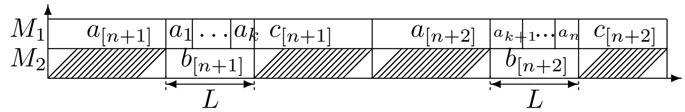

Fig. 2. The solution of the problem with $C_{max} = 6L + 4$

$C_{max}(\sigma) \leq 6L + 4$. As the sum of processing times of all tasks on the primary machine (M_1) is equal to $6L + 4$, that means that there is no idle time on this machine. The interlacement of the two tasks T_{n+1} and T_{n+2} is not possible because of their processing times on the first machine which is strictly greater than L ($a_j = c_j = L+1$). So, the tasks should be processed successively, as shown in Figure 2. These yield a solution of the partition problem.

Note that the symetric case is also NP-hard. (The proof is the same, we have just to replace a_j by c_j)

Theorem 2. $F2|chain - reentrant, no - wait, a_j \in \{0, L + 1\}, b_j \in \{0, L\}|C_{max}$ problem is NP-hard.

Theorem 3. $F2|chain - reentrant, no - wait, a_j = c_j \in \{0, L\}|C_{max}$ problem is NP-hard.

Proof.

Given an arbitrary instance of the partition problem, we consider the following instance of our scheduling problem, where the number of tasks is equal to $n + 2$ and the processing times are given in Table 2.
We seek a schedule σ of the $n+2$ tasks on the two machines M_1, M_2 with $C_{max}(\sigma) = 4L$?

Table 2. processing times of the tasks

$n+2 : tasks$	a_j	b_j	c_j
$T_j : j = 1, \ldots, n$	0	$b_j = v_j$	0
T_{n+1}	L	L	L
T_{n+2}	L	L	L

Our problem belongs to NP because we can verify in polynomial time if a permutation of tasks satisfies all the constraints of the problem.
We prove that the scheduling problem has a solution if and only if the partition problem has a solution.
If the partition problem has a solution, then there exists a partition of V into two disjoint subsets $V', V - V'$ such that $\sum_{j \in V'} v_j = \sum_{j \in V \setminus V'} v_j = L$. Suppose without loss of generality that $V' = \{v_1, \ldots, v_k\}$ and $V \setminus V' = \{v_{k+1}, \ldots, v_n\}$.
For this solution, we construct a solution of our problem as shown in Fig. 3.

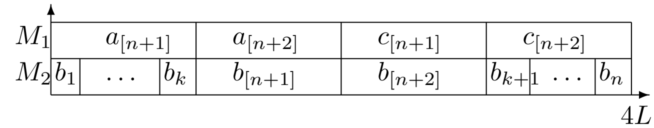

Fig. 3. The solution of the problem with $C_{max} = 4L$.

The value of the makespan is equal to $4L$.
Conversely, suppose there is a schedule σ of the scheduling problem with makespan $C_{max}(\sigma) \leq 4L$. As the sum of processing times of tasks on the primary machine M_1 is equal to $4L$, and the sum of processing times of tasks on