Computers & Industrial Engineering 56 (2009) 1494–1501

Contents lists available at ScienceDirect

Computers & Industrial Engineering

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

A hybrid artificial immune algorithm for a realistic variant of job shops to minimize the total completion time ☆

B. Naderi [a], M. Khalili [b], R. Tavakkoli-Moghaddam [c,*]

[a] Department of Industrial Engineering, Amirkabir University of Technology, Tehran, Iran
[b] Department of Industrial Engineering, Islamic Azad University, South Tehran Branch, Tehran, Iran
[c] Department of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran

ARTICLE INFO

Article history:
Received 27 January 2008
Received in revised form 7 September 2008
Accepted 12 September 2008
Available online 25 September 2008

Keywords:
Job shop scheduling
Total completion time
Sequence-dependent setup times
Flexible machine availability
Artificial immune algorithm

ABSTRACT

This paper investigates an extended problem of job shop scheduling to minimize the total completion time. With aim of actualization of the scheduling problems, many researchers have recently considered realistic assumptions in their problems. Two of the most applied assumptions are to consider sequence-dependent setup times and machine availability constraints (MACs). In this paper, we deal with a specific case of MACs caused by preventive maintenance (PM) operations. Contrary to the previous papers considering fixed or/and conservative policies, we consider flexible PM operations, in which PM operations may be postponed or expedited as required. A simple technique is employed to schedule production jobs along with the flexible MACs caused by PM. To solve the given problem, we present a novel meta-heuristic method based on the artificial immune algorithm (AIA) incorporating some advanced features. For further enhancement, the proposed AIA is hybridized with a simple and fast simulated annealing (SA). To evaluate the proposed algorithms, we compare our proposed AIA with three well-known algorithms taken from the literature. Finally, we find that the proposed AIA outperforms other algorithms.

1. Introduction

Job shop scheduling is one the most important scheduling environment taking place in many industrial setting. A job shop can be defined as follows: we have a set of n jobs that need to be processed on a set of m machines. Contrary to flow shops in which all the jobs have a same processing route, in job shops, it is assumed that each job j may have a unique processing route to meet the machines where O_{ij} denotes i-th machine that job j must be processed on. Each job can be processed by only one machine at a time and each machine can process only one job at a time. We consider a non-preemptive case meaning that the processing of a job cannot be interrupted. The buffer between every two machines is unlimited meaning that a job can wait limitlessly for a machine if that machine is occupied.

The aim is to find the job sequence on each machine in order to optimize the objective(s). The most frequently used objectives are the minimization of the makespan, maximum tardiness, total completion time, and total tardiness. The makespan and total completion time are manufacturer-oriented objectives while the maximum tardiness and total tardiness are costumer-oriented. In this paper, we intend to minimize the total completion time since it is regarded as a more realistic case of the makespan (Ruiz, Garica-Diaz, & Maroto, 2007). Since job shops belong to a specific class of combinatorial optimization problems known to be NP-hard ones (Cheung and Zhou, 2001), the presentation of metaheuristics is inevitable (Cheung and Zhou, 2001; Zhou, Beizhi, & Yang, 2006).

On the one hand, machine setup time is a significant factor for production scheduling in all flow patterns manufacturing environments (Ruiz, Maroto, & Alcaraz, 2005). Setups are usually performed between two consecutive jobs on the same machine. These setups are either sequence-independent or dependent setup times (SDST) (Monma & Potts, 1989). As general, sequence-independent setup times can be ignored or combined with the processing times. In many real-life situations, such as color procedure in plastic industry, wafer testing in semiconductor manufacturing, the setup operations, such as cleaning up or changing tools, are not only often required between jobs but they are also strongly dependent on the immediately preceding process on the same machine (Pinedo, 1995 and Sule, 1996). The main reason why researchers have been motivated to use this assumption is to solve scheduling problems in a real manner and also because of the tremendous savings when setup times are explicitly included in scheduling decisions. In addition, we assume that setup is non-anticipatory meaning that the setup can be only started as soon as the machine and the job are both available.

☆ This manuscript was processed by Area Editor Maged M. Dessouky.

* Corresponding author. Tel.: +98 2188021067; fax: +98 2188013102.
E-mail addresses: tavakoli@ut.ac.ir, tavakoli@mech.ubc.ca (R. Tavakkoli-Moghaddam).

0360-8352/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cie.2008.09.031

On the other hand, most papers dealing with scheduling problems assume that machines are always available during the scheduling period. However, in most real-life industrial cases, a machine can be unavailable for many reasons, such as unforeseen breakdowns (stochastic unavailability) or due to a scheduled preventive maintenance (PM) where the periods of unavailability are known in advance (i.e., deterministic unavailability). A breakdown makes the shop behavior hard to predict, and thereby reduces the efficiency of the production system. Therefore, scheduling maintenance to reduce the breakdown rate is commonly recognized by the decision makers. It is known that maintenance plays an important role in many industries, such as semiconductor and plastic industry; hence, it should be carefully explored. A poor scheduling of maintenance may greatly reduce the shop performance (Ruiz et al., 2007). As a result, the presentation of techniques to integrate production and PM activities is a key issue in the field of scheduling. Almost all the papers in the literature consider fixed or/and conservative policies (i.e., the PM operation must be scheduled at exactly predetermined intervals). We herewith apply a flexible criterion to consider PM operations along with productions jobs to gain more effective schedule. The problem studied in this paper can be denoted as $J/ST_{sd}/\sum C_j$ using the three-field notations of Graham, Lawler, Lenstra, and Rinnooy Kan (1979) to describe scheduling problems.

With reference to the above explanations, the aim of this paper is to propose a high performing algorithm for a realistic problem of job shops to minimize total completion times. To consider job shops with machine availability constraints (MACs), we employ a simple and flexible criterion to gain more effective schedule. In this paper, we apply a high performing meta-heuristic based on the concept of artificial immune algorithm (AIA). The reason to AIA's ever-increasing popularity among researchers is its powerful global exploration capability (Zandieh, Fatemi Ghomi, & Moattar Husseini, 2006). We also further enhance our proposed AIA by the hybridizing it with a very simple and fast form of simulated annealing (SA). We investigate its potential on solving the problem studied here against the adaptations of the some well-known algorithms in the literature through a set of instances.

The rest of the paper is organized as follows. The literature review of the problem is presented in Section 2. Section 3 describes how to consider flexile machine availability constraints into our problem. Section 4 introduces the proposed algorithms. Section 5 is devoted to the computational results. Finally, Section 6 gives some conclusions and future research.

2. Literature review

Coleman (1992) showed that the SDST single machine is strongly NP-hard and introduced an integer programming model to minimize the earliness and tardiness. As far as we concern, most papers in the literature of the scheduling problems with SDST focus on flow shops and its variants, such as hybrid flow shops, and flexible flow lines scheduling (Kurz & Askin, 2003; Kurz & Askin, 2004; Ruiz et al., 2005; Zandieh et al., 2006). Regarding SDST job shops, Brucker and Thiele (1996) proposed a branch-and-bound algorithm. Zhou and Egbelu (1989) presented a heuristic to minimize the makespan. Choi and Korkmaz (1997) considered job shops with the separable SDST under the minimization of makespan and modeled the problem as a mixed-integer programming. Cheung and Zhou (2001) proposed a genetic algorithm hybridized with a well-known dispatching rule for job shops where SDSTs are separable. The first operations for each of m machines are scheduled by GA while the next operations on each machine are scheduled by the SPT rule. Viond and Sridharan (2008) studied dynamic SDST job shops and presented a discrete event simulation model of the job shops. Zhou et al. (2006) proposed an immune algorithm for SDST job shops to minimize makespan. Allahverdi, Ng, Cheng, and Kovalyov (2006) provided a complete survey of scheduling problems with setup times.

A complete survey on existing algorithms to solve scheduling problems under availability constraints as well as complexity results is presented by Schmidt (2000). Holloway and Nelson (1974) implemented a multi-pass procedure in job shops by generating schedules periodically. Lee (1997) investigated the preemptive two-machine flow shops with one unavailability period first imposed on machine 1 and then on machine 2. To tackle the problem, he proposes some heuristics with error bounding analysis. Blazewicz, Breit, Formanowicz, Kubiak, and Schmidt (2001) considered a two-machine problem with an arbitrary number of unavailability periods on one machine. He proves that the minimization of makespan was strongly NP-hard. Breit (2006) studied a problem of two-machine scheduling with n preemptive jobs where the first machine is not available to process the jobs during a given time interval. Ruiz et al. (2007) considered flow shops with MACs caused by preventive maintenance policies. Naderi, Zandieh, and Fatemi Ghomi (2008a) studied a SDST flexible flow line with preventive maintenance and proposed a variable neighborhood search for the problem. As far as we concerned, production scheduling with MACs is mostly restricted to flowshops in the literature.

3. Flexible machine availability constraint

In a real industry, a machine may become unavailable during certain periods of time to process those jobs that are left in the previous horizon, breakdown, or preventive maintenance (PM) activities. Researchers usually consider PM activities as a common reason for the MACs. Many papers have studied to schedule the production jobs along with the PM operations. A commonly used PM policy is "*preventive maintenance at fixed predefined time intervals*" (Ruiz et al., 2007). The integrating criteria so far introduced are usually regarded as fixed and conservative plans (Naderi et al., 2008a; Ruiz et al., 2007). In the fixed plan, PM operations are carried out at exactly pre-determined time intervals while in the conservative plan, whenever production and PM operations have overlap, the production operation is postponed and PM operations is performed first.

In this study, we establish a more flexible criterion to integrate production scheduling and PM operations. That is, we assume that the starting time points of PM operations could be flexible to some extent (δ). In this case, likely more efficient schedules can be obtained. In a nutshell, our procedure of integration is as follows: let us suppose that the time interval between two consecutive PM operations is T_{PM}. Whenever a new job is to be processed in each machine, the completion time is computed. If this time exceeds the $T_{PM} + \delta$, then the process of the next job is postponed and the PM is carried out first. It is necessary to state that since we consider the non-preemptive case, the process of a job cannot be interrupted before it completes. To better clarify the above procedure, we apply it to an example. Let us consider a shop with $T_{PM} = 15$ time units. The duration of PM operations (D_{PM}) are three time units. The maximum accepted delay (δ) is four time units. The

Table 1
Processing times for a problem with $n = 5$

Job i	Processing time
1	5
2	9
3	6
4	7
5	4