Computers & Operations Research 65 (2016) 76–82

Contents lists available at ScienceDirect

Computers & Operations Research

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

Hybrid flow shop scheduling with not-all-machines options via local search with controlled deterioration

Deming Lei [a,*], Xiuping Guo [b]

[a] School of Automation, Wuhan University of Technology, Wuhan 430070, Hubei Province, PR China
[b] School of Economic and Management, Southwest Jiaotong University, Chengdu, Sichuan Province, PR China

ARTICLE INFO

Available online 11 July 2015

Keywords:
Hybrid flow shop scheduling
Local search
Controlled deterioration

ABSTRACT

In this study we consider hybrid flow shop scheduling problem with a decision referring to the number of machines to be used. A simple way is used to decide the number of the used machines. A novel local search with controlled deterioration (CDLS) is proposed, which is composed of multiple neighborhood searches with the prefixed number of iterations and deterioration step. The deterioration step tries to obtain a new current solution with the controlled deteriorated degree on the solution quality. CDLS is tested on a number of instances and the computational results show that CDLS can provide the promising results for the considered problem.

1. Introduction

Hybrid flow shop is a common manufacturing environment, which consists of a set of two or more processing stages with at least one stage having two or more parallel machines. The duplication of the number of machines in some stages can introduce flexibility, increase capacities and avoid bottleneck if some operations are too long. Hybrid flow shop has been applied in many industries such as chemical industry, spring and wire manufacturing, thus, hybrid flow shop scheduling problem (HFSP) has very great importance and attracted much attention from the theoretical and practical researchers.

HFSP has been extensively considered in the past 50 years. Ruiz et al. [1] summarize the type of HFSP addressed in more than 200 papers and describe many approaches including exact algorithm, heuristic and meta-heuristic for HFSP and its many variants. Generally, HFSP has at least two stages. As the simple version of HFSP, two-stage HFSP has been investigated by some researches. Figielska [2] proposes a heuristic for scheduling in the two-stage flow shop with one machine at the first stage and parallel unrelated machines at the second stage, where the renewable resources are shared among the stages. Rabiee et al. [3] consider no-wait two-stage HFSP with unrelated parallel machines and propose a hybrid algorithm based on imperialist competitive algorithm. Gerstle and Mosheiov [4] study a two-stage HFSP with a decision referring to the number of machines to be used and provide a closed form expression for the optimal number of machines to be used. Wang and Liu [5] present a multi-objective tabu search for two-stage HFSP with preventive maintenance. Wang and Liu [6] consider a two-stage HFSP with the dedicated machines and propose a heuristic method based on branch and bound (BB) algorithm. Wang and Liu [7] present a genetic algorithm (GA) for two-stage no-wait HFSP.

A number of papers have discussed HFSP with more than two stages and the various processing constraints and conditions are considered in these papers. Marichelvam et al. [8] develop an improved cuckoo search algorithm for multi-stage HFSP. Mirsanei et al. [9] propose a novel simulated annealing (NSA) algorithm for HFSP with sequence-dependent setup times. Dugardin et al. [10] present a new non-dominated sorting genetic algorithm with Lorenz dominance for a reentrant HFSP. Khalouli et al. [11] propose an ant colony optimization algorithm for HFSP considering the minimization of the total earliness and tardiness penalties. Amin-Naseri and Beheshti-Nia [12] consider the problem of parallel batch scheduling in a hybrid flow shop and present a three dimensional GA. Shiau et al. [13] present a hybrid method based on constructive GA for scheduling problem in proportionate flexible flow shop.

The previous studies on HFSP have the following features: (1) HFSP with the various processing constraints such as no-wait, maintenance and sequence-dependent setup times is considered fully. Two-stage HFSP also attract much attention; however, some processing constraints such as not-all-machines (NAM) options are seldom considered. (2) Some meta-heuristics such as GA and tabu search have been applied to solve HFSP; however, meta-heuristics are not applied to HFSP with NAM options.

When NAM options, which mean that it is not necessary to use all machines of each stage, are used, the scheduler is required to

* Corresponding author. Tel.: +8615327311013.
E-mail address: deminglei11@163.com (D. Lei).

http://dx.doi.org/10.1016/j.cor.2015.05.010
0305-0548/

decide the number of the used machines. Only several papers discuss NAM options in the scheduling problem. Cheng and Li [14] consider the case that jobs can be either processed in-house or outsourced to external production lines and decide the best number of outsourced machines to be employed. Finke et al. [15] and Kravchenko and Werner [16] also study NAM options. Gerstle and Mosheiov [4] first consider NAM options in flow shop environments. NAM options are not investigated fully in scheduling problems.

In the existing local search algorithms, both iterated local search (ILS, Lourenco et al. [17]) and simulated annealing (SA) accept the deteriorated solution to overcome local optimality. In ILS, perturbation step produces new starting points of local search by perturbing some solutions and the perturbed solutions are directly accepted even if the perturbed solutions are notably worse than the current ones. In SA, some deteriorated solutions are always accepted according to the acceptance probability. Obviously, the deteriorated degree in these two algorithms is not controlled. The uncontrolled deterioration can lead to new starting points; however, the uncontrolled deterioration may frequently lead to the current solution deteriorate and a useless cycle occurs very probably, which means the current solution is not improved, even deteriorates after many iterations. The useless cycle may be harmful to avoid falling local optima. The above feature of ILS and SA motivate us to propose a novel local search with the controlled deterioration (CDLS).

In this study, HFSP with NAM options is considered and CDLS is applied to solve the problem to minimize makespan. CDLS consists of multiple neighborhood searches with the prefixed number of iterations and deterioration step. The deterioration step tries to obtain a new current solution with the controlled deteriorated degree on the solution quality. A simple way is also proposed to decide the number of the used machines. CDLS is tested on a number of instances and the computational results show that CDLS is a very competitive method for the considered problem.

The remainder of the paper is organized as follows. Problem under study is described in Section 2. The proposed CDLS for the problem is shown in Section 3. Numerical test experiments on CDLS are reported in Section 4 and the conclusions are summarized in the final section and some topics of the future research are provided.

2. Problem description

HFSP is generally to optimize the processing of a set of n jobs in a series of $m(\geq 2)$ stages in terms of certain objectives. HFSP is quite common in practice, especially in the process industry. Each stage k has $\theta_k(\geq 1)$ identical machines in parallel and at least one stage has more than one identical machine in parallel. p_{ik} indicates the processing time of job $J_i(i=1,2,\cdots,n)$ at stage k. All jobs are processed following the same production flow: stage 1, stage 2, ……,stage m. A job can skip any number of stages provided it is processed in at least one stage. When a job J_i skip stage k, the processing of J_i is not done at stage k and p_{ik} must be equal to 0.

Table 1 shows an illustrative example of HFSP, in which the datum of each grid is the processing time of a job and symbol "-"means that the job is not processed at the corresponding stage, that is, the job skips the stage, for example, job J_1 skips stage 2.

In the most of the scheduling literature, all available machines are always used in the production systems; however, the situation, in which only a subset of the machines should be used, should be considered in many cases, for example, when the capacity exceeds the demand or the option of renting some machines to a third company is considered, not all machines should be employed. To the best of our knowledge, NAM options are first studied in hybrid flow shop by Gerstle and Mosheiov [4]; however, they just consider NAM options in two-stage HFSP.

In this study, HFSP with NAM options is considered which has the same constraints as HFSP. These constraints are listed below: all machines and jobs are available from time zero; each job can only be processed on one machine at a time and each machine can process no more than one job at a time; pre-emption is not allowed et al. Generally, HFSP is a NP hard problem. The considered problem is itself HFSP after the used machines are selected, so it is obvious that the computational complexity of the problem is also NP hard.

The goal of the problem is to obtain a schedule to minimize makespan

$$C_{\max} = \max_{i=1,2,\cdots,n} \{C_i\} \tag{1}$$

Where $C_{\max}$ and C_i are the makespan and the completion time of job J_i.

Fig. 1 shows the schedules of the example shown in Table 1. Two schedules have the same makespan; however, 9 machines are used in Fig. 1(a).

3. CDLS for HFSP with NAM options

As stated above, the uncontrolled deterioration of some local search algorithms such as ILS and SA may be harmful to obtain the optimal solution, it is reasonable to introduce the controlled deterioration into local search to avoid useless cycle, so we construct a CDLS for the considered problem.

3.1. Initial solution

For the considered problem, the decision on the number of the used machines and scheduling should be done jointly; however, if the number of used machines and scheduling are jointly optimized, when a new number of used machines is obtained, the schedule will be changed and should be adjusted many times to adapt to the new number of used machines, as a result, the current solution of CDLS is also changed, in some cases, the current solution will deteriorate and the search may be stagnate; moreover, it is easier to decide a reasonable number of the used machines than to obtain a schedule, thus we first decide the number of the used machines, then we mainly focus on scheduling using CDLS.

Table 1
An illustrative example of the problem.

Job	Stage 1 M_1, M_2, M_3	Stage 2 M_4, M_5, M_6, M_7	Stage 3 M_8, M_9, M_{10}	Job	Stage 1 M_1, M_2, M_3	Stage 2 M_4, M_5, M_6, M_7	Stage 3 M_8, M_9, M_{10}
J_1	71	–	96	J_4	94	49	55
J_2	77	35	50	J_5	76	94	78
J_3	30	74	65	J_6	82	41	96