Heuristic algorithms for permutation flow shop scheduling problem with deteriorating jobs

Mohammad Ranjbar
Department of Industrial Engineering, Faculty of Engineering, Ferdowsi University of Mashhad
Mashhad, Iran

m_ranjbar@um.ac.ir

Sanaz Sharifi Ghazvini
Department of Industrial Engineering, Faculty of Engineering, Ferdowsi University of Mashhad
Mashhad, Iran

snz.sharifi@stu-mail.um.ac.ir

Abstract

Scheduling with deteriorating jobs has received increasing attention in the few last years. In this paper, permutation flow shop scheduling problem with deteriorating jobs is considered. Processing time of each job is considered as an increasing and linear deterioration function of its starting time. The objective is to schedule each activity such that the makespan is minimized. Three heuristic procedures including the modified slope heuristic, the minimum waiting times heuristic and the ratio heuristic are developed to obtain high quality solutions. Computational results on randomly generated test sets indicate that the ratio heuristic algorithm and the minimum waiting time heuristic algorithm are more efficient than the modified slope heuristic.

1. INTRODUCTION

The research in flow shop scheduling problem has resulted in many useful models and algorithms over the last decades ([3]). A very common goal in flow shop scheduling problems is minimization of the makespan, i.e., the maximum completion time of the jobs. This problem is denoted as $Fm||C_{max}$ in the notation of [4], and has been proved to be NP-hard by [2] when more than two machines are considered. Several heuristic and exact procedures have been proposed to solve this problem. In most of these researches, it is assumed that job processing times are fixed and known in advance while deteriorating jobs exists in practice. Such deterioration appears in, e.g., scheduling of maintaining jobs, modeling of fire fighting, cleaning assignments, etc (See [10]). In order to model these practical problems, processing time of each job is considered as a function of its starting time and its position in sequence. In other words, the jobs processed later consume more time than the jobs processed earlier. The scheduling problem with deteriorating jobs was introduced by [5] in a steel rolling mill and [7] also applies the concept of deteriorating jobs in fire fighting. They assume that control of fire in a special place is considered as a job and processing time of this job depends on its starting time because if the fire is not controlled soon, it will be expanded immediately.

Most of researches in scheduling with deteriorating jobs are related to the single machine problems. In contrast of single machine case, there are a few articles in flow shop scheduling problem with deteriorating jobs. [6] considers the makespan minimization and showed under linear deterioration, two-machine flow shop scheduling is strongly NP-Hard. [8] introduces a polynomial algorithm for two-machine flow shop scheduling and proved when an arbitrary number of machine is assumed, it is a NP-hard problem. Moreover, [10] considers flow shop scheduling problem with deteriorating jobs under dominating machines and aims to minimize the makespan.

In this research, we consider the permutation flow shop scheduling problem with deteriorating jobs where deterioration function is linear. The objective is to minimize the makespan. Three heuristics and fast procedures are developed and compared in terms of efficiency.

The permutation flow shop scheduling problem consists of scheduling n independent and non-preemptive jobs gathered in set $J = \{J_1, J_2, \dots, J_n\}$ on m machines $M = \{M_1, M_2, \dots, M_m\}$. All jobs should follow a fixed route of machines to be completed and due to the permutation assumption, sequence of jobs on all machines are identical. We assume the processing time P_{ij} of job j ($j = 1, \dots, n$) on machine i ($i = 1, \dots, m$) is given as a linear function of its starting time t, i.e. $P_{ij}(t) = \alpha_{ij}t + \beta_{ij}$ where $\alpha_{ij} \geq 0$ denotes the deterioration rate of job j on machine i and $\beta_{ij} \geq 0$ is the fixed process time of job j on machine i. It is also assumed that all jobs are available for processing at the beginning of the planning horizon. There are also unlimited intermediate storages between successive machines. The completion time of job j on machine i is shown by C_{ij} and the objective is to minimize the makespan, C_{max}, where $C_{max} = max\{C_{ij}\}$. On the basis of the standard notations of [4], this problem can be denoted as $Fm|prmu, P_{ij}(t) = \alpha_{ij}t + \beta_{ij}|C_{max}$.

2. THE HEURISTIC ALGORITHMS

In this section, we develop three heuristic procedures to deal with $Fm|prmu, P_{ij}(t) = \alpha_{ij}t + \beta_{ij}|C_{max}$ including modified slope heuristic algorithm (MSHA), minimum waiting time heuristic algorithm (MWTHA) and ratio heuristic algorithm (RHA). In order to show the efficiency of three previous developed heuristic procedures, we also have used a randomized heuristic algorithm (RNHA).

2.1. Modified slope heuristic algorithm

The main idea of MSHA is based on the slope algorithm, developed by [9] for $F_m|prmu|C_{max}$. The MSHA can be described by the following steps.

*Step*1: let $A_j(t)$ as the slope of each job $j \in J$ and calculate its value as $A_j(t) = -\sum_{i=1}^{m}\big(m - (2i-1)\big)P_{ij}(t) = a_j t + b_j$.

*Step*2: let $h_j = \begin{cases} -a_j b_j\ ;\ a_j < 0 \text{ and } b_j < 0 \\ a_j b_j; Otherwise \end{cases}$ for every $j \in J$.

*Step*3: Arrange jobs in decreasing order of h_j indices.

Based on the idea of the Johnson's rule, the jobs with larger b_j index should be processed earlier than the jobs with smaller b_j index. On the other hand, the jobs with larger a_j index will have a big processing time if they are processed later. Thus, the jobs with larger values of $h_j = a_j b_j$ has higher priority than other jobs. An exception

occurs if both a_j and b_j are negative. In this case, $h_j = -a_j b_j$ is conceptually correct.

2.2. Minimum waiting times heuristic algorithm

The main idea of the MWTHA is taken from the profile fitting algorithm, developed by [1] for $Fm|block|C_{max}$. In the MWTHA the sequence of jobs is determined such that the waiting time for jobs and machines is minimized. The waiting time for job $j = 2, \dots, n$ on machine $i; i = 2, \dots, m$, shown by $wt_j^{'i}$, is positive if $c_{i-1,j} < c_{i,j-1}$. In this case, $wt_j^{'i} = c_{i,j-1} - c_{i-1,j}$. Also, the waiting (or waste) time on machine $i; i = 2, \dots, m$ between jobs $j-1$ and j ; $j = 2, \dots, n$, shown by $wt_j^{''i}$, is positive if $c_{i-1,j} > c_{i,j-1}$. In this case, $wt_j^{''i} = c_{i-1,j} - c_{i,j-1}$. We define $wt_j = \sum_{i=2}^{m} \left(wt_j^{'i} + wt_j^{''i} \right)$. The MWTHA can be described by the following steps.

*Step*1. Select the job with the smallest sum of fixed processing times to go first on all machines. Remove this job from the set of remaining unscheduled jobs.

*Step*2. To determine which job should go second, every remaining unscheduled job is tried out. For each candidate job *j*, calculate wt_j. The job with the smallest wt_j is selected as the next job in sequence. Remove the selected job from the set of remaining unscheduled jobs.

*Step*3. If the set of remaining unscheduled jobs is empty, stop, otherwise; go to step2.

2.3. The ratio heuristic algorithm

The main idea of the RHA is that we prefer to process jobs with larger deterioration rate and smaller fixed processing time earlier than other jobs. For this purpose, the subsequent steps should be followed.

*Step*1: calculate the rate r_{ij} for every job on each machine as $r_{ij} = \frac{\alpha_{ij}}{\beta_{ij}}$.

*Step*2: calculate the total rate tr_j for every job $j \in J$ as $tr_j = \sum_{i=1}^{m} r_{ij}$.

*Step*3: arrange job based on decreasing order of tr_j indices.

2.4. Random heuristic algorithm

In RNHA, we simply generate a random sequence of jobs and construct its corresponding schedule

3. COMPUTATIONAL EXPERIMENTS

In order to evaluate efficiency of the developed algorithms, 31 test instances were generated. The test instances were taken from the OR-Library (http://people.brunel.ac.uk/~mastjjb/jeb/info.ht ml) which were developed for the $Fm||C_{max}$. The problem size varies between 7 and 100 jobs and between 4 and 20 machines. In order to customize the test instances for the $Fm|prmu, P_{ij}(t) = \alpha_{ij} t + \beta_{ij}|C_{max}$, we considered the fixed and available processing time as the β_{ij} values and we generated α_{ij} values randomly from uniform distribution $U[0,1]$.

Summary and comparative computational results are presented in Table 1 in which two numbers are shown in each cell. The first one indicates the average percent deviation from best found solutions and the second one, shown inside parenthesis, displays the number of best solutions found by the corresponding algorithm. In terms of the average percent deviation, the RHA has the best performance while MWTHA, MSHA and RNHA are in the next orders. In addition, the MWTHA could find the maximum number of best solutions while the RHA, MSHA and RNHA are in the next orders based on this criterion.

Table 1. Comparative computational results

MSHA	MWTHA	RHA	RNHA
48.46 (6)	41.50(14)	38.99(9)	57.12(2)

4. CONCLUSIONS

In this paper, we proposed four heuristic algorithms to solve flow shop scheduling problem with deteriorating jobs and makespan criterion. We also assumed that the jobs processing times follow a linear function of deterioration rate (P_{ij} (t) = α_{ij} t + β_{ij}). Computational results showed that the ratio heuristic algorithm outperforms other developed heuristic algorithms.

As a future research opportunity, we distinguish shop scheduling with deteriorating jobs in more complicated scheduling environments (e.g. job shop, open shop); this would constitute an important step also towards the possibility of practical implementation of the models. The development of other heuristic, meta-heuristic or exact approaches for the problem considered in this paper may also be interesting research topics.

REFERENCES

[1] Mc Cormic, M. Pinedo, S. Shenker, B. Wolf, *Sequencing in an assembly line with blocking to minimize cycle time*, Operations Research, vol. 37, pp. 925-936, 1989.

[2] M.R. Garey, D.S. Johnson, *Commuters and intractability: A guided tour to the theory of NP-completeness*, San Francisco: Freeman, (1979).

[3] J. Grabowski, M. Wodecki, *A very fast tabu search algorithm for the permutation flow shop problem with makespan criterion*, Computers & Operations Research, vol. 31, pp. 1891-1909, 2004.

[4] R.L. Graham, E.L. Lawler, J.K. Lenstra, A.H.G. RinnooyKan, *Optimization and Approximation in deterministic sequencing and scheduling theory: a survey*, Annals of Discrete Mathematics, vol. 5, pp. 287-326, 1979.

[5] J.N.D. Gupta, S.K. Gupta, S.K. *Single facility scheduling with nonlinear processing times*, Computers & Industrial Engineering, vol. 14, pp. 387 393, 1988.

[6] A. Kononov, S. Gawiejnowicz, *NP-hard cases in scheduling deteriorating jobs on dedicated machines*, Journal of Operations Research Society, vol. 52, pp. 708–717, 2001.

[7] A.S. Kunnathur, S.K. Gupta, *Minimizing the makespan with late start penalties added to processing times in a single facility scheduling problem*, European Journal of Operational Research, vol. 47, pp. 56–64, 1990.

[8] G. Mosheiov, *Complexity analysis of job-shop scheduling with deteriorating jobs*, Discrete Applied Mathematics, vol. 117, pp. 195–209, 2002.

[9] D.S. Palmer, *Sequencing jobs through a multi-stage process in the minimum total time- a quick method of obtaining a near optimum*, Operational Research Quarterly, vol. 16, pp. 101-107, 1965.

[10] J.B. Wang, *Flow shop scheduling with deteriorating jobs under dominating machines to minimize makespan*, *International Journal of Advanced Manufacturing Technology*, vol. 48, pp. 719–723, 2010.