Mathematics

مجلة كلية التربية
العدد الثالث /2006

Exact and Near Optimal Solution for Three Machine Flow Shop Scheduling

Dr. Tariq S. Abdul-Razaq *Manal H. AL-Harby*
Dept. of Mathematics , College of Science,
University of AL-Mustansiriyah

Abstract

In this paper we consider F3/ /C_{max} problem in which the processing order is assumed to be the same on each machine, the resulting problem is called the permutation flow shop problem. The objective is to find a processing order on each machine such that the makespan C_{max} , which is the completion time of the last job is minimized. This problem is NP-hard, hence the existence of a polynomial time algorithm for finding an optimal solution is unlikely. This complexity result leads us to use an enumeration solution approach. In this paper we propose a branch and bound algorithm to solve this problem. Also we develop fast approximation algorithms yielding near optimal solution. We

report on computation experience; the performance of exact and approximation algorithms are tested on large class of test problems.

1. Introduction

In the flow shop scheduling problem, we are giving machines $M_1, M_2, \ldots, M_m$ where $m \geq 2$,and a set $N=\{1,2,\ldots,n\}$ of jobs .
Each job has to be processed first on M_1, then on M_2, and so on, until it is processed on the last machine M_m. The processing time P_{ij} of each job $i \in N$ on each machine M_j $(1 \leq j \leq m)$ is given. Preemption is not allowed. Each machine processes at most one job at a time , and each job is processed on at most one machine at a time. The objective function to be minimized is the C_{max} , i.e., the maximum completion time of all jobs on all machines . For a schedule s, the value of the C_{max} is denoted by C_{max} (s). A schedule that minimizes the C_{max} is called optimal and is denoted by s*.

Following standard notation (Lawler et al. 1993) , we denote this problem by Fm// C_{max} or by F// C_{max} if m is variable.

Consider the calculation of C_{max} for the Fm// C_{max} problem. Let s be an arbitrary permutation of n jobs. For simplicity assume s=1,2,...,n. It is well known that the completion time C_i^k of job i (of sequence s) on machine M_k is given by [3]

$$\left.\begin{aligned} & C_i^k = \max_i\{C_i^{k-1}, C_{i-1}^k\} + P_{ik} \\ & C_i^\circ = C_\circ^k = 0 \qquad \forall 1 \leq i \leq n, \quad 1 \leq k \leq m \end{aligned}\right\} \qquad (1)$$

where P_{ik} is the processing of job i on M_k , k=1,...,m.

It is well known THAT(Conway et al. 1967) [5], Rinnooy Kan 1976 [12], Lenstra (1977) [8], they see that to find the optimum Therefor the Fm//C_{max} problem, we need to consider only schedules with the same processing order on the first two machines and the same processing order on the last two machines. Therefore it is well known that for both problem F2//C_{max} and F3//C_{max} there exists an optimal solution that is a permutation schedule for which all machines process the jobs according to the same job sequence (Conway et al .1967)[5] . However, for $Fm// C_{max}$, when $m \geq 4$, it can be the case that no optimal solution is a permutation schedule(Conway et al.) [5] . The following two- job $F4// C_{max}$ example ,given in Conway et al.[5], shows that this result can not be extended any further.

Let $P_{11}= P_{22}= P_{23}= P_{14}= 4$, $p_{21}= p_{12}= p_{13}= p_{24}= 1$. There are only two order preserving schedules (see Fig.(1)) both of which have a maximum completion time of 14.

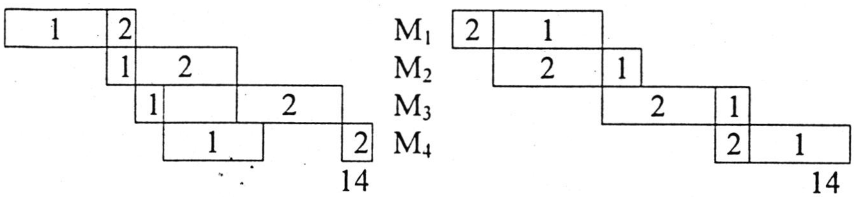

Figure (1)

Now consider a schedule(see Fig.(2)) which has the same processing order on machines M_1 and M_2 and the same processing order on machines M_3 and M_4 but in which the order is reversed between machines M_2 and M_3. The maximum completion time is 12, which is less than what was obtained above.

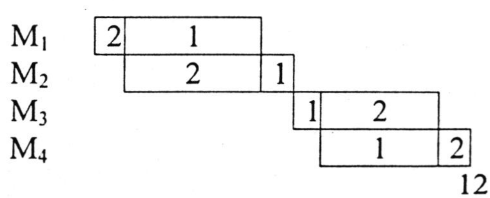

Figure (2)