Applied Mathematical Modelling 40 (2016) 4231–4247

Contents lists available at ScienceDirect

Applied Mathematical Modelling

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

Optimal robot scheduling to minimize the makespan in a three-machine flow-shop environment with job-independent processing times

Dvir Shabtay*, Kfir Arviv

Department of Industrial Engineering and Management, Ben-Gurion University of the Negev, Beer-Sheva, Israel

A R T I C L E I N F O

Article history:
Received 21 March 2015
Revised 29 July 2015
Accepted 9 November 2015
Available online 22 November 2015

Keywords:
Scheduling
Flow-shop
Identical jobs
Makespan
Transportation times
Robot

A B S T R A C T

We study a robotic three-machine flow-shop scheduling problem, in which n identical jobs are to be processed and the objective is to minimize the makespan. After the job's completion on either the first or the second machine it is transferred by a robot to the next (consecutive) machine in the shop. A single robot is available for transferring the jobs. We show that the problem can be solved by decomposing it into a set of sub-problems, and by providing a robot schedule to each sub-problem that yields a makespan value which matches the lower bound value.

1. Introduction

In this paper we study a flow-shop scheduling problem in which jobs are identical and a single robot is responsible for the transportation of jobs between consecutive machines. Our objective is to provide a robot schedule so as to minimize the makespan. Robotic flow-shop systems are widely prevalent in automatic manufacturing systems (see, e.g., Crama and Van de Klundert [7], Zhou et al. [49] and Yan et al. [47]), and therefore providing an efficient robot and machine schedule for such systems is an important challenge. Several different sets of robotic flow-shop scheduling problems have been considered in the literature; those most frequently studied are listed below.

- *Set 1*: a set consisting of a single robot, where the objective is to minimize the makespan. There may be two types of capacity constraints. The first relates to the capacity of the input and output buffers of each machine, and the second relates to the robot capacity, i.e., a constraint on the number of jobs that the robot can transfer in a single move. Different problems belong to this set have been analyzed, among others, by Stern and Vinter [44], Panwalkar [39], Kise [26], Kise et al. [27], Levner et al. [33], Hurink and Knust [20], Lee and Chen [29], Lee and Strusevich [31], Tang and Liu [45] and Ling and Guang [35].
- *Set 2*: a set of problems consisting of a sufficient number of robots with no technological constraints such that any job that is completed on any one of the machines is immediately transferred to the input buffer of the next (consecutive) machine in the shop with no delays (see, e.g., Maggu and Das [36], Yu [48], Dell'Amico [15] and Karuno and Nagamochi [24]).

* Corresponding author. Tel.: +972 8 6461389; fax: +972 8 6472958.
E-mail addresses: dvirs@bgu.ac.il (D. Shabtay), arvivk@bgu.ac.il (K. Arviv).

http://dx.doi.org/10.1016/j.apm.2015.11.021

- *Set 3*: a set in which the production is cyclic and thus the objective is to minimize the cycle time, i.e., to maximize the production rate (see, e.g., Sethi et al. [41], Batur et al. [4], Crama and Van de Klundert [6] and [7], Karaznov and Livshits [25], Gultekin et al. [19], Agnetis [1], Kats and Levner [21], Levner et al. [34], Che et al. [8–10], Chu [13], Kats et al. [22], Zhou et al. [49] and Lei et al.[32]).

Our problem belongs to the first set of problems, so we begin by first defining this set of problems and then provide a brief literature review. A general definition of the first set of problems can be stated as follows: A set of n jobs, $\mathcal{J} = \{J_1, \ldots, J_n\}$, is available for processing at time zero, and is to be scheduled on a set of m machines, $\mathcal{M} = \{M_1, \ldots, M_m\}$, in a flow-shop scheduling system. In such a system, each job J_j consists of m operations $\mathcal{O}_j = \{O_{1j}, \ldots, O_{mj}\}$, which must be processed in the order $O_{1j} \to \ldots \to O_{mj}$. Operation O_{ij} must be processed on machine M_i without preemption for $p_{ij} \geq 0$ time units. Each machine M_i $(i = 1, \ldots, m)$ has both an input buffer and an output buffer, I_i and O_i, with a capacity of c_{I_i} and c_{O_i}, respectively, meaning that the inventory of jobs in these buffers is limited to c_{I_i} and c_{O_i} units, respectively. It is assumed that there is an automatic mechanism beside each machine M_i which allows each robot to perform both download and upload operations from the buffers in negligible time. A single robot is responsible for the transportation of any job J_j $(j = 1, \ldots, n)$ from each machine M_i to its consecutive machine M_{i+1} $(i = 1, \ldots, m-1)$. Let t_{ij} be the transportation time required for the robot to transfer job J_j from machine M_i to machine M_{i+1}, and let te_i $(i = 1, \ldots, m-1)$ be the time required for the robot to return empty (without carrying a job) from machine M_{i+1} to machine M_i. The number of jobs that can be transferred in a single move is limited to be not greater than c_R, which is the robot capacity limitation. Moreover, the empty return times are assumed to be additive, i.e., the time for the robot to travel between two distinct machines is the sum of the empty traveling times between all intermediate machines. For simplicity, we omit the machine index when $m = 2$, such that, $t_{1j} = t_j$ is the transfer time of job J_j from machine M_1 to machine M_2, and the empty return time from M_2 to M_1 is simply te. For a given robot and machine schedule, let C_j be the completion time of job J_j for $j = 1, \ldots, n$ on M_m. Our objective is to find an optimal robot and machine schedule that minimizes the makespan, which is defined by $C_{\max} = \max_{j=1,\ldots,n}\{C_j\}$.

Throughout this paper we will use the standard three-field notation, $\alpha|\beta|\gamma$, introduced by Graham et al. [18] for scheduling problems. The α field presents the machine environment and contains a single entry. We include an $Fm, R1$ entry in the α field to denote a flow-shop scheduling system with m machines and a single robot. The β field describes the processing characteristics and constraints. For example, if $p_{ij} = p_i$ $(t_{ij} = t_i)$ is specified in this field, it implies that the job processing times (transportation times) are job-independent, and if $c_{I_i} = c_I$ is specified, it implies that the size of the input buffer is machine-independent. The γ field contains the optimization criteria, which is the makespan $(C_{\max})$. Unless explicitly stated elsewhere in the β field, we assume an unlimited capacity of input and output buffers (i.e., that $c_{I_i} \geq n$ and $c_{O_i} \geq n$ for $i = 1, \ldots, m$), and that the robot can only move a single job at a time (i.e., $c_R = 1$). For example, $Fm, R1|t_{ij} = t_i|C_{\max}$ refers to a makespan minimization flow-shop scheduling problem with m machines and a single robot that is responsible to transfer jobs between machines. The non-empty transportation times are machine-dependent and job-independent. There is no capacity limitation on the buffers between the machines, and the number of jobs to be transferred in a single robot move is restricted to one.

1.1. Literature review

Kise [26] proves that the $F2, R1|t_j = t|C_{\max}$ problem is ordinary $\mathcal{NP}$-hard. Hurink and Knust [20] study various variants of the $Fm, R1||C_{\max}$ problem with zero empty return times ($te_i = 0$ for $i = 1, \ldots, m-1$). They prove that the $F2, R1|te_i = 0|C_{\max}$ problem is equivalent to the well-known $F3||C_{\max}$ problem, and is thus strongly $\mathcal{NP}$-hard. They also prove that the less general problems $F2, R1|t_j = t, te_i = 0|C_{\max}$ and $F2, R1|p_{ij} = p, te_i = 0|C_{\max}$ are strongly $\mathcal{NP}$-hard, and that other special cases of the $F2, R1|te_i = 0|C_{\max}$ problem can be solved in polynomial time. These include the special case where either *all* processing times are restricted to unity, and the case of equal processing times with only two possible transportation times. Hurink and Knust further show that if all processing times are equal and the transportation times are job-independent (but machine-dependent) then the resulting $Fm, R1|p_{ij} = p, t_{ij} = t_i, te_i = 0|C_{\max}$ problem can be solved in polynomial time for *any* arbitrary number of machines. Ling and Guang [35] show that the $F2, R1|p_{ij} = p_j, t_j \in \{t_1, t_2\}, te_i = 0|C_{\max}$ problem is strongly $\mathcal{NP}$-hard, where $t_j \in \{t_1, t_2\}$ implies that there are only two possible transportation times. In all the above-mentioned papers it is assumed that the robot has the capacity to transfer a single job at a time (that is, $c_R = 1$). Lee and Chen [29] study the case where the robot has the capacity to transfer more than a single job in a single move. They show that the $F2, R1|t_j = t, c_R \geq 3|C_{\max}$ problem is strongly $\mathcal{NP}$-hard, and that problems $F2, R1|p_{1j} = p_j, t_j = t, c_R = \bar{c}_R|C_{\max}$ and $F2, R1|p_{2j} = p_j, t_j = t, c_R = \bar{c}_R|C_{\max}$ are solvable in polynomial time for any fixed capacity of the robot, $\bar{c}_R$. Lee and Strusevich [31] study the $F2, R1|c_R \geq n|C_{\max}$ problem, where $c_R \geq n$ indicates that the capacity of the robot is unlimited. They consider a family of schedules $\mathcal{F}(2)$ that includes only two non-empty robot moves from M_1 to M_2, which further implies that there is only a single empty move from M_2 to M_1. They prove that the problem of finding the best schedule in $\mathcal{F}(2)$ is $\mathcal{NP}$-hard, and that for any $\varepsilon > 0$ there is an instance of the problem for which the makespan of the best schedule from class $\mathcal{F}(2)$ is strictly greater than $3/2 - \varepsilon$ times the optimal value. Then, they construct a heuristic algorithm that finds a schedule from class $\mathcal{F}(2)$ which is not more than 3/2 times the value of the optimal solution. Stern and Vinter [44] study the $F2, R1|c_{I_i} = c_{O_i} = 0|C_{\max}$ problem, reformulate it as a asymmetric traveling salesman problem, and provide a heuristic procedure for its solution. Later, Ganesharajah et al. [16] prove that this problem is strongly $\mathcal{NP}$-hard by a reduction from the well-known strongly $\mathcal{NP}$-hard $F3|no\text{-}wait|C_{\max}$ problem. Other variants of the problem on two machines have been studied by Panwalkar [39], Kise et al. [27], Levner et al. [33] and Tang and Liu [45].