Int. J. Production Economics 141 (2013) 286–294

Contents lists available at SciVerse ScienceDirect

Int. J. Production Economics

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

Two-machine flow shop scheduling of polyurethane foam production

Bertrand M.T. Lin*, Y.-Y. Lin, K.-T. Fang

Institute of Information Management, Department of Information and Finance Management, National Chiao Tung University, Hsinchu 300, Taiwan

ARTICLE INFO

Article history:
Received 23 June 2011
Accepted 1 August 2012
Available online 16 August 2012

Keywords:
Polyurethane foam
Flow shop
Precedence constraints
Branch-and-bound algorithm
Heuristic
Iterative local search

ABSTRACT

This paper studies a two-machine flow shop scheduling problem with a supporting precedence relation. The model originates from a real production context of a chemical factory that produces foam-rubber products. We extend the traditional two-machine flow shop by dividing the operations into two categories: supporting tasks and regular jobs. In the model, several different compositions of foam rubber can be mixed at the foam blooming stage, and products are processed at the manufacturing stage. Each job (product) on the second machine cannot start until its supporting tasks (parts) on the first machine are all finished and the second machine is not occupied. The objective is to find a schedule that minimizes the total job completion time. The studied problem is strongly NP-hard. In this paper, we propose a branch-and-bound algorithm incorporating a lower bound and two dominance rules. We also design a simple heuristic and an iterated local search (ILS) algorithm to derive approximate solutions. The performances of the proposed algorithms are examined through computational experiments.

1. Introduction

This research investigates a flow shop scheduling model inspired by a real production line of polyurethane (PU) foam at a manufacturing site in central Taiwan. Due to different chemical compositions, various types of foams, including general-PU foam, inert foam, viscoelastic (VE) foam and bamboo charcoal foam, can be produced by mixing different materials on a foam blooming machine. While only one foam blooming machine is available, a certain amount of each composition type can be processed at a time. When a composition is finished, the foam can be segmented or sliced into specific sizes for different final products on another machine. To synthesize a final product (job), different types of compositions could be required. Consider the following example production scenario. There are four products to be produced—multi-layer mattress, single-layer mattress, memory pillow and seat pad. Materials required for producing the above products include general PU foam, inert foam and bamboo charcoal foam. The combination of materials for the four products is shown below:

Product 1 Multi-layer mattress requires general-PU foam and inert foam;
Product 2 Single-layer mattress requires general PU foam;
Product 3 Memory pillow requires inert foam;
Product 4 (Seat pad) requires general PU foam and bamboo charcoal foam.

Fig. 1 depicts a production sequence (Product 2, Product 3, Product 1, Product 4). Note that Product 1 cannot be produced until the processing of general PU foam and inert foam is finished at the first stage.

The above foam production environment can be modeled as a two-machine flow shop. Johnson's seminal work (1954) has spurred extensive research works on flow shop scheduling with new manufacturing settings and different objective functions (El-Bouri et al., 2008; Fondrevelle et al., 2009; Haouari and Hidri, 2008; Haq et al., 2010; Yang, 2010). A flow shop consists of several machines arranged in series, and each stage consists of a single machine such that all jobs or products must visit the machines along the specified route. To minimize the time required for finishing all jobs in a two-machine flow shop, Johnson (1954) proposed an elegant algorithm that can solve the problem in polynomial time. In a two-machine flow shop, each job (product) has two operations to process on the machines subject to the specified route, i.e. all jobs need to visit the first machine and then the second machine. Moreover, each operation on the second machine cannot start until the corresponding operation on the first machine is finished and the second machine is not occupied. The problem studied in this research is an extension of flow shop scheduling in the following aspects: For a specific product, it requires one or more types of foams. Preparation of the required foams is performed on the first machine, while production of the final products is carried out

* Corresponding author. Tel.: +886 3 5729915; Fax: +886 3 5131472
E-mail address: bmtlin@mail.nctu.edu.tw (B.M.T. Lin).

0925-5273/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ijpe.2012.08.006

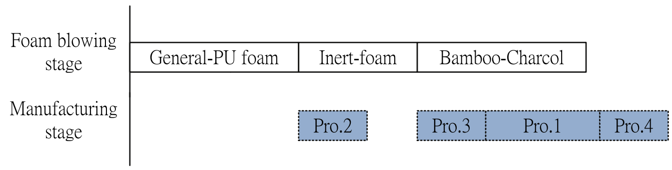

Fig. 1. Gantt chart of an example production schedule.

on the second machine. The production process of a product on the second machine cannot start until the second machine is available and all the required foams are prepared and ready for use. The production model in this research exhibits a clear difference from traditional two-machine flow shop scheduling. A multiple-to-multiple relation exists between machine-one operations and machine-two operations because a type of foam can support one or more products and vice versa. This feature exhibits a significant difference from the one-to-one relationship inherent in the traditional two-machine flow shop. In summary, in the studied model all operations are categorized into two types: *supporting tasks* and *regular jobs*. Machine one is dedicated to the supporting tasks and machine two to the regular jobs. A job can be processed only if the second machine is free and all of its supporting tasks have been done on the first machine. The model was also studied by Chen and Lee (2009) in the context of cross-docking to minimize the makespan of the jobs on the second machine. This paper will investigate the scheduling problem of minimizing the sum of job completion times on the second machine, in short, *the total job completion time*. This objective function reflects not only the service quality, indicating the average customer waiting time, but also the work-in-process inventory cost.

The rest of this paper is organized as follows. Section 2 presents formal statements of the problem definition as well as reviews related previous works. In Section 3, we will propose an integer linear programming model and some preliminary properties that will assist the development of solution algorithms. Section 4 is dedicated to the development of a lower bound and two dominance rules that are to be included in a branch-and-bound algorithm for solving the problem optimally. To derive production schedules in an acceptable time, two approximation algorithms, including a greedy heuristic and an iterated local search (ILS) algorithm, are designed in Section 5. A computational study on the proposed algorithms is given in Section 6. We give conclusions and suggest potential research directions in Section 7.

2. Problem statements and literature review

This section first gives formal statements of the studied problem. Notation and an example follow. Review on related works will also be presented.

The scheduling problem is formally defined as follows: There are two disjoint sets of operations $A=\{a_1,a_2,\ldots,a_m\}$ and $B=\{b_1,b_2,\ldots,b_n\}$ to process on two machines M_A and M_B, respectively. Processing times of $a_i \in A$ and $b_j \in B$ are denoted by p_i^a and p_j^b, respectively. The relation between the operations of sets A and B is specified by the supporting relation $\mathcal{R} \subseteq A \times B$ such that for $a_i \in A$ and $b_j \in B$, if $(a_i,b_j) \in \mathcal{R}$ then operation b_j cannot start on machine M_B unless operation a_i is complete on machine M_A. Hereafter, we call the elements of set A (*supporting*) tasks and the elements of set B (*regular*) jobs. Let α_i denote the subset of jobs supported by task i, and β_j the subset of tasks supporting job j. For example, if $\mathcal{R}=\{(a_1,b_1),(a_1,b_2),(a_2,b_1),(a_3,b_2)\}$ then $\alpha_1=\{1,2\},\alpha_2=\{1\},\alpha_3=\{2\}$, and $\beta_1=\{1,2\},\beta_2=\{1,3\}$.

The uniqueness of the studied problem lies in the virtual but mandatory role of supporting operations of set A. The setting of machine M_A and machine M_B can be treated as a flow shop. The sum of processing times of the tasks in β_j on machine M_A corresponds to the processing time of job b_j on the first machine in a traditional two-machine flow shop, where relation $\mathcal{R}$ is a one-to-one and onto function, or in other words, $|\alpha_i|=1$ for all tasks a_i and $|\beta_j|=1$ for all jobs b_j. This is depicted in Fig. 2. In the studied problem, the supporting tasks of a job are not always processed consecutively on machine M_A. Similarly, the jobs supported by a task are not required to be executed consecutively on machine M_B, either. Therefore, the structure of the problem setting is much more complicated. This paper investigates the objective function of the total completion time.

Throughout the paper, a sequence of tasks on machine M_A is denoted by $s=(s_1,\ldots,s_m)$, and a sequence of jobs on machine M_B by $S=(S_1,\ldots,S_n)$. In a particular schedule, C_i^A denotes the completion time of task a_i on M_A, and C_j^B the completion time of job b_j on M_B. Function $Z(s,S)$ gives the objective value under sequences s and S. Later, we will show that parameter s can be omitted for it can be determined once a job S is given.

To illustrate the problem definition, we consider the following instance. There are five tasks $A=\{a_1,a_2,a_3,a_4,a_5\}$ and four jobs $B=\{b_1,b_2,b_3,b_4\}$. The processing times are shown below.

tasks	a_1	a_2	a_3	a_4	a_5
p_i^A	6	3	2	5	9

jobs	b_1	b_2	b_3	b_4
p_j^B	10	3	1	7

The supporting relation is $\mathcal{R}=\{(a_1,b_1),(a_1,b_3),(a_2,b_1),(a_2,b_4),(a_3,b_2),(a_3,b_3),(a_4,b_3),(a_5,b_4)\}$, as depicted in Fig. 3(a). Given sequence s=(3, 4, 1, 2, 5) on M_A and sequence S=(2, 3, 1, 4) on M_B, we have the corresponding Gantt chart shown in Fig. 3(b). The completion times of the jobs on machine M_B are 5, 14, 26 and 33. Therefore, the total completion time is 78.

Since Johnson's paper (1954), flow shop scheduling has been widely studied in the literature. While the minimization of makespan in a two-machine flow shop ($F2||C_{\max}$) can be solved by Johnson's $O(n \log n)$-time algorithm, the minimization of total completion time ($F2\|\sum C_j$) is nevertheless strongly NP-hard (Garey et al., 1976). With the strongly NP-hard $F2\|\sum C_j$ problem as a special case, the problem we are considering is also computationally intractable. To the best of our knowledge, there are two models related to the production setting of this paper. First, Chen and Lee (2009) studied a cross-docking problem where a warehouse receives goods from various vendors and then repackages the goods for distributions to various destinations. The operations can be regarded as the two-machine flow shop setting investigated in this paper. They proved the problem of makespan minimization to be strongly NP-hard, proposed a branch-and-bound algorithm, and designed a heuristic algorithm. It is shown that the ratio between the heuristic solution and optimal solution is not greater than 3/2. The second related model is due to Lin et al. 2010 where all of the supporting tasks and the regular jobs are processed on a single machine. The model stems from streaming and scheduling of multi-media objects. They discussed the complexity status of three objective functions, namely $L_{\max}$, $\sum w_jC_j$, and $\sum w_jU_j$, on the single-machine setting and extended the existing complexity results of single-machine scheduling with precedence constraints.