Computers & Industrial Engineering 75 (2014) 142–156

Contents lists available at ScienceDirect

Computers & Industrial Engineering

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

Minimizing total earliness and tardiness on a permutation flow shop using VNS and MIP ☆

R. M'Hallah *

Department of Statistics and Operations Research, College of Science, Kuwait University, P.O. Box 5969, Safat 13060, Kuwait

ARTICLE INFO

Article history:
Received 5 May 2013
Received in revised form 27 March 2014
Accepted 16 June 2014
Available online 26 June 2014

Keywords:
Scheduling
Flow shop
Variable neighborhood search
Earliness
Tardiness
Local search

ABSTRACT

This paper addresses the NP hard earliness tardiness m-stage permutation flow shop scheduling problem where idle time can be inserted. It proposes different new formulations for the problem, and provides computational proof of the superiority of the positional model. This latter yields when solved with a mixed integer programming solver the exact solution for small or easy instances. For large and difficult instances, the paper proposes an approximate approach $\mathcal{H}$ that hybridizes variable neighborhood search (VNS) with mixed integer programming (MIP). VNS searches for the best sequence of the jobs whereas MIP inserts idle time optimally for each sequence. In addition, $\mathcal{H}$ feeds the VNS near global optimum and its value to the solver of the positional model. They constitute a good initial solution and a valid upper bound. Extensive experimental investigation highlights the usefulness of the hybridization and the competitiveness of $\mathcal{H}$. This hybrid approach can be easily extended to more complex scheduling problems.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

The globalization of the world economy along with the widespread of internet have shifted the production strategy of many industrial sectors from massive production for a single client to make-to-order for a multitude of customers scattered all over the world. To remain competitive and offer a sustainable advantage, many small and middle-sized companies find themselves planning their production on a day-by-day basis as a function of the small-quantity orders they receive. Their competitive advantage is to meet the target delivery date of each customer regardless of the size of the order or its destination. Most apparel industries in developing countries within a close proximity to financially-attractive markets strive to retain this advantage.

Meeting the delivery dates of customers translates to minimizing the sum of earliness and tardiness of the orders. In fact, when completed early, an order causes unwanted inventory and/or deterioration of the product. When an order is tardy, a customer may cancel it or request a rebate, or the producer may resort to expediting the shipment. Thus, the length an order is tardy reflects customer satisfaction and the length it is early measures inventory performance.

In many apparel industries, the production line is similar to a flow shop where each order (or job) undergoes a preset sequence of operations with a known deterministic processing time for each operation (or stage). This particular type of industries falls into a larger group of production environments that are analogous to a flow shop or its flexible variant. This group encloses manufacturing of electronics, oil products, cars and airplanes, along with many types of service industries (Behnamian & Zandieh, 2011; Haouari and M'Hallah, 1997; Pinedo, 2002).

This paper tackles the m-machine flow shop scheduling problem with distinct due dates. A set $N = \{1, \ldots, n\}$ of n jobs, available at time zero, are to be scheduled successively on each of a set $M = \{1, \ldots, m\}$ of m machines. A job $j \in N$ is characterized by its positive integer known due date d_j and processing time p_{ij} on machine $i \in M$. All n jobs are ready at time zero and job preemption is not allowed. Furthermore, there is sufficient buffer space between the successive machines. That is, there is no machine blocking and there is no constraint on the waiting time of a job between successive stages. The objective is to identify a schedule that minimizes $\sum_{j=1}^{n}(E_j + T_j)$, the sum of earliness and tardiness (ET) of the n jobs, where $E_j = \max\{0, d_j - C_j\}$ and $T_j = \max\{0, C_j - d_j\}$ are the earliness and tardiness of job $j \in N$, and C_j its completion time on the flow shop. Job j is early if it

☆ This manuscript was processed by Area Editor Hans Kellerer.

* Tel.: +965 669 14150; fax: +965 248 37332.
E-mail address: rymmha@yahoo.com

http://dx.doi.org/10.1016/j.cie.2014.06.011
0360-8352/© 2014 Elsevier Ltd. All rights reserved.

completes before d_j, on-time if it completes on d_j, and tardy otherwise.

This paper focuses on the case where machines can be idle. Cases where idle time cannot be inserted do exist in real life settings. Such cases are generally motivated by the prohibitively high cost of idling the machines, the technological infeasibility of idling them for repetitive short periods of time, or the reduced availability of the technicians operating them. However, inhibiting idle time appears contradictory with the idea of minimizing total earliness and tardiness (Sarper, 1995). As in the single machine case, inserting idle time may reduce total earliness and tardiness (Baker & Scudder, 1990). When inserting idle time is technologically feasible, it is sufficient to insert idle time on the last machine of the flow shop problem (Hendel & Sourd, 2007; Sarper, 1995).

This problem, denoted as $Fm|d_j|\sum E_j + T_j$, is NP hard since it is an extension of the NP hard $1|d_j|\sum T_j$ and $1|d_j|\sum E_j + T_j$ (Lawler, 1977). Thus, finding an exact solution via mathematical programming approaches is only possible for easy or small-sized instances. For large and difficult instances, commercial solvers encounter difficulties in reaching the optima. For these instances, the paper proposes a hybrid heuristic $\mathcal{H}$, which combines variable neighborhood search (VNS) and mixed integer programming (MIP).

VNS is a general framework for designing meta heuristics. Its prevalence is due not only to its simplicity and few parameters, but mainly to its effective and efficient results. It applies iteratively – deterministically or stochastically – the two key factors for the success of a cooperative algorithm: exploitation (or intensification) and exploration (or diversification).

$\mathcal{H}$ uses a VNS variant $\mathcal{V}$ whose objective is to find, among all possible sequences, the permutation $\overline{\pi}$ that minimizes ET. $\mathcal{V}$ is fed with a permutation of the n jobs. It perturbs this permutation obtaining a sequence π' and undertakes a steepest descent in the neighborhood of π'. It changes the focal point and size of the neighborhood when it identifies an improving solution, and alters the neighborhood type when trapped in a local optimum.

$\mathcal{H}$ applies a low-level hybridization to $\mathcal{V}$. It computes ET(π), the minimal ET of a sequence π, via an MIP model that inserts idle time on the last machine optimally. In addition, it employs a high-level hybridization. It initiates a branch and bound search from the best solution $\overline{\pi}$ obtained by $\mathcal{V}$.

Section 2 reviews solution approaches for the ET flow shop and VNS applications to scheduling. Section 3 proposes a new mathematical model for $Fm|d_j|\sum E_j + T_j$ along with adaptations of existing models to the problem. Section 4 details the proposed heuristic $\mathcal{H}$. Section 5 displays the computational results highlighting the advantage of the proposed mathematical model and the good performance of the solution approach. Finally, Section 6 summarizes the results and suggests future areas of research.

2. Literature review

Section 2.1 gives a brief overview of recent work on the earliness tardiness flow shop problem. Section 2.2 motivates the use of VNS, defines its most general algorithm, and lists some of its applications to scheduling.

2.1. Earliness tardiness flow shop problems

The first investigated ET problem (Kanet, 1981) schedules a set of jobs with a common due date on a single machine. Subsequent research extends the problem to different machine environments with a (n) (un) known common due date. Lauff and Werner (2004a) provide a detailed literature survey on the common due date ET multiple machine scheduling, comparing different approaches and pinpointing their limitations. They further elucidate the confusion regarding restrictive and unrestrictive due dates and the impact of the definition on the difficulty of the problem and on the validity of the proposed algorithms. They state that Sarper (1995) was the first to extend the common due date ET problem to a multiple-stage machine environment; specifically, to the two-machine flow shop. In a related paper, Lauff and Werner (2004b) study the characteristics of the two-stage common due date scheduling problem (including the flow shop case) where the due date is nil or non-restrictive (i.e., does not impact the scheduling decision). Their objective is to minimize the sum of ET and waiting time of jobs between the two machines. Min and Sung (2001) consider the two-machine flow shop problem when the common due date is larger than the makespan of the first machine and jobs are processed in batches. They study the properties of three classes of problems where the jobs of a batch have identical processing times. Chandra, Mehta, and Tirupati (2009) present a precedence-based mixed integer program for the ET permutation flow shop problem where all jobs have a common due date. In addition, they classify the problem in terms of the tightness of the due date and provide the characteristics of the optimum for each class.

However, in many real life settings such as the apparel industry, the assumption of a common due date is not realistic. Few authors drop this assumption. Yet, they restrict their attention to the case where no idle time is inserted; that is, no machine is kept idle when a job is present for processing. Zegordi, Itoh, and Enkawa (1995) approximately solve the weighted ET (WET) m-machine flow shop via a simulated annealing based heuristic where a job is moved forward or backward into the sequence if it has a higher priority index than its right or left adjacent neighbor. They simplify the computation of the priority index by basing it on an estimate of the job's earliest starting time on machine m. They test their approach on instances with up to 30 jobs and 20 machines. The large number of machines is only pertinent because it assesses the quality of the estimation of the priority index. Madhushini, Rajendran, and Deepa (2009) present a breadth first branch and bound algorithm for scheduling permutation flow shops for a variety of objectives including minimizing WET. The algorithm fathoms nodes using job-based lower bounds and machine-based upper bounds on the completion time of a job. Schaller and Valente (2013) propose a genetic algorithm for the ET m-machine permutation flow shop. They compare the performance of their GA to existing heuristics that were designed for related flow shop and single machine problems.

Variants of the problem have similarly been pondered. Janiak, Kozan, Lichtenstein, and Oguz (2007) present three heuristics based on simulated annealing and tabu search for the hybrid flow shop problem with release dates where the objective is to minimize the sum of WET and waiting time between successive stages of all jobs. Khalouli, Ghedjati, and Hamzaoui (2010) propose an ant colony optimization heuristic for the ET hybrid flowshop problem. Behnamian and Zandieh (2011) address the hybrid flowshop with limited waiting times of jobs between successive stages and sequence-dependent set up times. They minimize the sum of earliness and squared tardiness using a colonial competitive algorithm. Moslehi, Mirzaee, Vasei, and Azaron (2009) minimize the sum of maximum earliness and tardiness using a branch and bound tree whose size is reduced via developed dominance properties. Huang, Huang, and Lai (2012) design a modified random key GA for a WET flowshop problem where the processing times and due dates are fuzzy.

The no-inserted idle time assumption is realistic in many instances. However, Sarper (1995) argues that it is in discord with the ET objective. Thus, this paper focuses on the $Fm|d_j|\sum E_j + T_j$ where it is technologically feasible to insert idle time. It identifies the best sequence using a VNS variant and inserts idle time using an exact solution to an MIP.

Download English Version:

https://daneshyari.com/en/article/1134033

Download Persian Version:

https://daneshyari.com/article/1134033

Daneshyari.com