Computers & Industrial Engineering 63 (2012) 120–134

Contents lists available at SciVerse ScienceDirect

Computers & Industrial Engineering

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

Solving two-machine assembly scheduling problems with inventory constraints ☆

Daria Terekhov [a], Mustafa K. Doğru [b,1,2], Ulaş Özen [b,1], J. Christopher Beck [a,*]

[a] Department of Mechanical and Industrial Engineering, University of Toronto, 5 King's College Road, Toronto, Ontario, Canada M5S 3G8
[b] Alcatel-Lucent Bell Labs, Blanchardstown Industrial Park, Dublin 15, Ireland

ARTICLE INFO

Article history:
Received 6 April 2011
Received in revised form 28 October 2011
Accepted 6 February 2012
Available online 16 February 2012

Keywords:
Assembly scheduling
Inventory
Order scheduling
Mixed-integer programming
Constraint programming

ABSTRACT

This paper considers a scheduling problem with component availability constraints in a supply chain consisting of two manufacturing facilities and a merge-in-transit facility. Three mixed-integer programming (MIP) models and a constraint programming (CP) model are compared in an extensive numerical study. Results show that when there are no components shared among the two manufacturers, the MIP model based on time-index variables is the best for proving optimality for problems with short processing times whereas the CP model tends to perform better than the others for problems with a large range of processing times. When shared components are added, the performance of all models deteriorates, with the time-indexed MIP providing the best results. By explicitly modelling the dependence of scheduling decisions on the availability of component parts, we contribute to the literature on the integration of inventory and scheduling decisions, which is necessary for solving realistic supply chain problems.

1. Introduction

Effective supply chain planning consists of solving many complex inter-related problems involving, for example, routing, inventory management and scheduling. Each of these problems is typically difficult to solve even when one ignores the connections between the different levels of decision-making and different facilities. For instance, many deterministic scheduling problems occurring within a single facility are known to be NP-hard (Pinedo, 2003). For some real life situations, modelling the relationships between different facilities may be important, but this increases the problem complexity further. In this paper, we focus on a problem arising in supply chains in which the goal is to construct production schedules at the manufacturing facilities while taking into account local constraints and the objective of the overall supply chain.

We consider a supply chain consisting of two manufacturing and a merge-in-transit (MIT) facilities. We assume that all facilities have the same owner and that there is a centralized decision-maker who can optimize the performance of the overall system. Adopting a high level view of the problem, we model the two manufacturing facilities as unary-capacity machines producing *sub-assemblies* which are then sent to an assembly machine that represents the MIT facility. In other words, a *sub-assembly* is assumed to be composed of the necessary quantity of products belonging to a customer order that need to be processed at a particular manufacturing facility. The processing times for each sub-assembly are determined by the quantity and configuration of products that are requested by the customer. We model the MIT facility as an infinite-capacity assembly resource (or, equivalently, as a single machine with a negligible processing time for each customer order). A customer order cannot be processed by this assembly machine unless the two sub-assemblies have both been completed. Fig. 1 is a schematic representation of this setup.

To be manufactured, each sub-assembly requires component parts, the mix and quantity of which are dependent on the type and quantity of products making up the sub-assembly. The components may be unique to a sub-assembly, or may be shared among all or a subset of the sub-assemblies scheduled on a machine or on both machines. Fig. 1 depicts the different categories of components as labeled triangles: triangle 1 represents all components that are consumed only at machine 1, which includes components unique to particular sub-assemblies processed on machine 1, and components shared among sub-assemblies on machine 1; triangle 2 represents all components used only at machine 2; triangle 0 corresponds to the components shared among the two machines. All components are replenished periodically at known points in time.

Each customer order has a due date and a weight for representing the penalty the company has to pay if the order is not completed before the due date. Given a set of customer orders, each consisting of two sub-assemblies with specific processing times

---

☆ This manuscript was processed by Area Editor T.C. Edwin Cheng.

* Corresponding author. Tel.: +1 416 946 8854; fax: +1 416 978 7753.

E-mail addresses: dterekho@mie.utoronto.ca (D. Terekhov), mustafa.dogru@alcatel-lucent.com (M.K. Doğru), ulas.ozen@alcatel-lucent.com (U. Özen), jcb@mie.utoronto.ca (J.C. Beck).

[1] Partly supported by Industrial Development Agency (IDA), Ireland.

[2] Present address: Alcatel-Lucent Bell Labs, 600 Mountain Avenue, Murray Hill, NJ 07974, USA.

0360-8352/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cie.2012.02.006

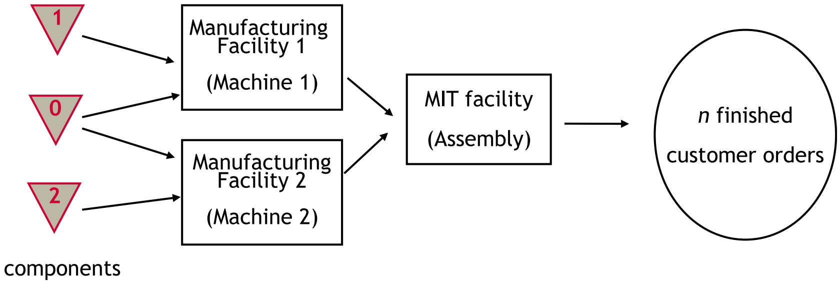

Fig. 1. Schematic representation of the assembly scheduling problem.

and component requirements, the goal of the problem is to schedule each sub-assembly on the corresponding manufacturing machine so as to minimize the total weighted order tardiness.

The model in our study is a stylized representation of a scheduling problem seen in the supply chain of Alcatel-Lucent – a global corporation providing telecommunications equipment and solutions to service providers, enterprises and governments. Network deployment projects often generate customer orders that are composed of products from different families (e.g., base stations, routers) manufactured in different plants or at different production lines in the same facility. Finished products are shipped to the nearest MIT facility (typically a regional warehouse) close to the customer, where they are merged into a single order with the necessary ancillary equipment like cables and connectors, and the order is shipped out to the customer location as a whole.

The aim of our study is to develop and evaluate modelling approaches that practitioners can implement on commercially available optimization software without resorting to problem specific algorithms. We present three mixed-integer programming (MIP) models for the problem under consideration: two of them, the *timeIndexed* model and the *positionalVariables* model, are extensions of single-machine MIP models from the literature (Keha, Khowala, & Fowler, 2009) and one (the *KolischModel*) is a specialization of the MIP model proposed by Kolisch and Hess (2000). We also propose a constraint programming (CP) model for the problem. In an extensive numerical study, we test the performance of these four models in terms of proving optimality and finding a feasible solution within a given run-time limit. For problems with components unique to sub-assemblies or shared among sub-assemblies processed on the same machine, the *timeIndexed* model is the best performer when processing times are short, while the CP model outperforms the others for problems that have a large range of processing times. When components are shared between the two machines, the problem becomes more difficult to solve: the best method, the *timeIndexed* model, proves optimality in only 18% of instances.

The main contributions of this work are:

1. We provide three MIP models and a CP model for solving a realistic scheduling problem with inventory constraints and an assembly structure. All of these models can be implemented using commercially-available software.
2. We experimentally demonstrate the effectiveness of these models for problems without shared components.

This paper is organized as follows. We provide an overview of the relevant literature in Section 2. In Section 3, we present a detailed description of the problem. Three MIP models are given in Section 4 and a constraint programming model is stated in Section 5. In Section 6, experimental results are presented. Section 7 discusses possible ideas for future work. Section 8 concludes the paper.

2. Literature review

Our problem has two characteristics that make it particularly complex: (i) an assembly structure relating several machines and (ii) component availability constraints.

2.1. Assembly structure among machines

The first characteristic has been studied in the literature on scheduling with assembly operations. In the absence of component availability constraints and under a high level view, our problem can be formulated as one of the following problems in the literature: order scheduling, assembly flow shop scheduling or assembly job shop scheduling.

In *order scheduling* (Lin & Kononov, 2007; Leung, Li, & Pinedo, 2005c), orders are assumed to be composed of m operations, each requiring a machine. Operations on different machines may be performed concurrently, and an order is complete when all m operations have been performed. The scheduling problem in such an environment has also been referred to as the *concurrent job shop scheduling problem* (Roemer, 2006), the *problem of scheduling customer orders* (Ahmadi & Bagchi, 1990), the *open shops with jobs overlap problem* (Leung, Li, Pinedo, & Sriskandarajah, 2005a) and *scheduling with bundled operations* (Li & Vairaktarakis, 2007). The order scheduling literature considers scheduling with identical parallel machines (Yang & Posner, 2005), any of which can process any product of an order, and with dedicated machines (Li & Vairaktarakis, 2007; Leung, Li, & Pinedo, 2007; Sung & Yoon, 1998). Of interest in this paper is the dedicated machine environment, denoted by PDm.[3] The literature includes work on complete and heuristic approaches, as well as on the development of theoretical properties. A MIP formulation for the minimum total weighted completion time problem is given by Ahmadi, Bagchi, and Roemer (2005). This formulation is very similar to the *positionalVariables* model presented in Section 4.2. More recent work on the concurrent open shop problem includes the paper by Mastrolilli, Queyranne, Schulz, Svensson, and Uhan (2010) as well as a book chapter by Huang and Lin (2007, Chap. 12).

The *assembly flow shop* literature deals with a more general problem: it explicitly models the assembly operation that is necessary for combining the products produced by the first-stage parallel operations. Initial work in this area was done by Lee, Cheng, and Lin (1993), based on two first-stage machines and the objective of minimizing makespan, C_{max}, which is defined as the completion time of the last scheduled job on the assembly machine. Potts

---

[3] In scheduling, the notation for describing a problem is $\alpha|\beta|\gamma$, where α denotes the machine environment, β denotes job characteristics and γ is the objective function. An empty α field implies a single-machine environment, while an empty β refers to the fact that preemptions are not allowed (Pinedo, 2003). Thus, $PD2||\sum C_j$ describes an order scheduling problem with two dedicated machines and the goal of minimizing the sum of completion times.