Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computers & Operations Research 32 (2005) 1213–1236
computers & operations research
www.elsevier.com/locate/dsw

Aggregate production planning for a continuous reconfigurable manufacturing process

Akhil Jain, Udatta S. Palekar*

Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

Abstract

Most of the research on aggregate production planning has been focused on discrete parts manufacturing models. In environments where intermediate inventory cannot be stored, and multiple products are produced simultaneously using complex configurations of production machines, these models may produce erroneous results. In this paper, we present a *configuration-based formulation* for one such manufacturing environment, where production may involve dissimilar machines performing similar operations at different rates and equipment can be connected together to form different production lines. The production process is continuous and no in-process inventory can be kept. We present and compare several heuristics to generate input data to solve the aggregate production-planning problems using the *configuration-based formulation*. Computational experiments show that large-scale real-world problems we encountered can be solved in reasonable time using our heuristics and commercial optimization software like CPLEX.

Keywords: Aggregate planning; Integer programming; Continuous production

1. Introduction

The aggregate production-planning problem has been studied extensively, since it was first formulated in the 1950s. Interest in the problem stems from the ability such models provide to control production and inventory costs. The costs associated with inventory management and production planning are a substantial portion of the total expenditure of the manufacturing companies. On average, 44.2 percent of net fixed assets of US manufacturing firms and 16.3 percent of their total

* Corresponding author. Fax: +1-217-244-6534.
E-mail addresses: akhjain@hotmail.com (A. Jain), palekar@uiuc.edu (U.S. Palekar).

0305-0548/$ - see front matter
doi:10.1016/j.cor.2003.11.001

assets were invested in inventory in 1986 (US Federal Trade Commission 1986, pp. 4–5). The value of inventories in US firms was about $1.1 trillion in 1993. The number for Canada was approximately $135 billion in 1994 [1]. IIE Solutions [2] reports that in only a few years, typical spare parts inventory had increased from $2–10 million to $5–15 million, an increase of at least 50 percent. Significant savings can be realized by correctly modelling and solving the aggregate production-planning problem.

Most of the research on this problem has been focused on models for discrete parts manufacturing. These models typically permit storage of in-process inventory between different stages of manufacturing. Moreover, parts are assumed to be independently processed on dedicated machines at every stage of manufacturing. However, such models do not accurately reflect the true production capacity when multiple products are co-manufactured using complex configurations of production machines. The problem is further exacerbated when the production process involves multiple stages and in-process inventory is not permitted.

In this paper, we consider one such situation, in which a base product is differentiated into multiple products as it passes through several production stages. The process is a continuous production process and no in-process inventories may be kept. Several machines are available for each stage of production and these machines can be combined in different ways to create machine configurations capable of co-manufacturing groups of products at various rates. These machines may be reconfigured, incurring a setup cost, into different configurations, to produce different sets of products. The rate at which a machine produces a product therefore depends on the configuration as well as the products involved. We call this problem the continuous process with dissimilar machine (CDM) aggregate planning problem.

The CDM problem arises in a number of contexts. Our model was designed for a large food-processing company. The company has numerous production facilities of varying sizes. The number of machines at each facility ranges between 3 and 57 machines and each facility produces between 2 and 300 stock-keeping units (SKUs). Our model can also find extensive application in the chemical industry, where in-process inventories cannot be stored because of cost considerations or short shelf life of the intermediate products. The general aggregate production-planning problem has been studied since the 1950s. Dzielinski and Gomory [3], Lasdon and Terjung [4] and Newson [5] have developed mathematical programming models for the problem of aggregate production planning, but for the case where these production stages can be handled independently, i.e., storage of intermediates is allowed. McClain et al. [6], Nahmias [7] and Silver et al. [1] also consider the problem of aggregate production planning for production processes where the production stages can be decoupled. Nam et al. [8] present a survey of the literature on aggregate production planning since early 1950s. They categorize the various existing techniques into a framework depending upon their abilities to produce an exact optimal or near-optimal solutions. Note that all these models allow the storage of intermediates and all machines are assumed to be identical.

Although there is a vast amount of literature discussing production-planning models (e.g., see Thomas and McClain [9] for an overview of production planning and Shapiro [10] for mathematical programming models for production planning), few models address the CDM problem. Leachman and Carmon [11] develop a model for identifying capacity limitations on the machines for the above problem. Bradley and Arntzen [12] address the problem of simultaneous planning of capacity investment, inventory and production schedule, in a scenario where all machines do not perform similar operations. Bermon and Hood [13] developed a model for *IBM* where unrelated machines