

A Multistage Benders Decomposition Method for Production Planning Problems with Approved Vendor Matrices

L.H. Lee, E.P. Chew, and T.S. Ng, Dept. of Industrial and Systems Engineering, National University of Singapore, Singapore

Abstract

A production planning problem is studied based on an actual manufacturer of hard-disk drives that offers the *approved vendor matrix* as a competitive advantage. An approved vendor matrix allows customers to pick and choose the component suppliers for individual components or pairs of components constituting their product. The problem is to develop production plans that minimize the total tardiness in fulfilling customer orders while observing the matrix restrictions and limited component supplies. It is first shown that this problem has an equivalent multicommodity network flow representation. A solution procedure using multistage Benders decomposition is then developed. The computational efficiency of the approach is compared with the column-generation method and the CPLEX general-purpose LP solver under different scenarios of matrix restrictions.

Keywords: Production Planning, Approved Vendor Matrix, Network Flows

1. Introduction

A production planning problem is considered based on an actual hard-disk drive manufacturer. The hard-disk drive is basically an assembly of a number of important components, and for this manufacturer there are several vendors supplying each component. As a competitive advantage, the manufacturer allows customers to specify their own *approved vendor matrix* (AVM). The AVM lets the customers pick and choose preferred vendors for individual components or combinations of components in their product. In this work, an end-product is defined as the final assembly of components. A *build type* is the set of all end-products that uses the same combination of component vendors. Customers specify their demand in the form of order types. A build type can be assigned as a customer's order type only if it complies to the AVM specified by the customer. A summary of the problem is as follows. At the beginning of each planning horizon, there is

a set of customer demands due in any period. Demand that is unfulfilled by the due date is backlogged and charged with a tardiness penalty until it is fulfilled. Demand that is unfulfilled by the end of the planning horizon is penalized as a shortage. Production resources like manpower availability and component supply schedules cannot be changed within the planning horizon. The decisions to be made are which build types to produce in each period and their corresponding levels, and how to pack them for different customers, that is, how much of each build type to assign for each order type in order to achieve the minimum total backlog and shortage costs. This is the *Build-Pack Problem* first introduced by Lee, Chew, and Ng (2005).

The chief characteristic of the build-pack problem is the high proliferation of build types and the complexity of production planning imposed by the AVM. A hierarchical planning approach (Hax and Meal 1975; Graves 1982) is unsuitable because the set of order types overlaps the set served by each build type. Chu (1995) developed a myopic decomposition heuristic for a production problem that allows customers to specify preferred suppliers for individual components, but still does not explicitly address the problem of high build type proliferation. Lee, Chew, and Ng (2005) formulated the build-pack problem as a total tardiness problem (Wang 1995). The column-generation technique is then used to solve the model for an optimal build-and-pack schedule.

This work first shows that the column-generation approach can be arrived at by considering the build-pack problem as a multicommodity network flow (MCNF) problem. Furthermore, the MCNF model for the build-pack problem has a special multistage network structure. The primary motivation of this work is to study the performance of a solution ap-

proach that is based on this multistage network structure. Secondly, the motivation of presenting this solution and modeling approach is practical one, based on the sensible question of whether, in a situation where there are very little AVM restrictions, it is possible to avoid a formulation such as that in Lee, Chew, and Ng (2005), which explicitly identifies the build types. In the case when there are no AVM restrictions, the build-pack problem collapses into a very simple total tardiness planning problem that can be solved efficiently. The solution, which can be viewed as a master schedule of the production levels for each customer order, would then be sufficient because it would not be necessary to explicitly identify the different build types. It would, hence, be of value if this basic result can be used in some way when AVM restrictions are present. Some scheme is then required to disaggregate the master schedule solution into build schedules, and then to repair the solution if AVM restrictions are violated. The issue of interest would then be comparing the economy of the effort required to perform the repairing versus the column-generation approach.

The following notation is used for the inputs and parameters of the build-pack planning problem in this work:

- t production period, $t = 1 \cdot \cdot \cdot T$
- p product component
- k order type
- v component vendor
- d_t^k demand level for k due in t
- g_t^k tardiness cost per unit of k in period t
- r_k^p units of p to build per unit of k
- $m_{v,t}$ units of v arriving in period t
- P set of all components p
- V_p set of all vendors of component $p \in P$
- V_p^k set of all vendors of $p \in P$ that is acceptable in the AVM of k
- K set of all k
- K_v set of all $k \in K$ that can use vendor $v \in V_p$ for component $p \in P$ to make the final product

$$\Gamma_{k,v,v'} = \begin{cases} 1 & \text{if for } k, \text{ vendor } v \text{ of component } p \text{ cannot} \\ & \text{be used together with vendor } v' \text{ of} \\ & \text{component } p', \text{ where } v \in V_p, v' \in V_{p'} \forall p, \\ & p' \in P. \\ 0 & \text{otherwise.} \end{cases}$$

The next section is a case study of a manufacturer of hard-disk drives, which provides the background for this work. Section 3 presents a MCNF representation of the build-pack planning problem. Section 4 gives a multistage formulation of the problem and develops a solution procedure that uses the Benders decomposition approach. Computational results are presented in section 5 to draw comparisons with the well-known column-generation method for solving MCNF problems. Conclusions are given in section 6.

2. Case: Production Planning of Hard-Disk Drives

2.1 Background

This work is motivated by a problem faced by an actual manufacturer of hard-disk drives, whose customers are largely OEMs and reputable PC makers. This manufacturer purchases key components from multiple vendors on a long-term contract basis. It then assembles, tests, and packs the drives for the customers. The hard-disk drive company competes in a thin-margin, commoditized market, and to defend profit margins the company offers *customer flexibility* and *on-time delivery* as its competitive advantage. One scheme to implement customer flexibility is to enable customers to pick and choose preferred vendors for individual or combinations of components constituting their hard-disk drive.

The problem addressed starts with the release of the Master Production Schedule (MPS), which is a schedule of order types (by demand quantity and due date) to be fulfilled in the current week. The MPS is used to drive even more detailed schedules, which are implemented at the assembly plant. A *build plan* schedules the run quantities of build types in each period, while a *pack plan* assigns the build types to order types to fulfill the MPS.

Once the build and pack plans are generated, the rest of the production process is relatively straightforward. At the beginning of each production period, production supervisors refer to the build plans to draw components from the parts store, and these components are fed into the manufacturing cells. In the workcells, the components are assembled into the build types and are then passed to the test cells for software coding and power-up tests. Finally, the drives are labeled and packed for the customers as specified in the pack schedule and are shipped out

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات