A Multicommodity Flow Model for Manufacturing Planning Over Alternative Facilities

S. David Wu
Hakan Golbasi
Lehigh University

Report No. 98T-004

IMSE Technical Report 98T-004, Lehigh University
Submitted to Management Science

A Multicommodity Flow Model for Manufacturing Planning over Alternative Facilities

S. David Wu and Hakan Golbasi

Manufacturing Logistics Institute, Department of IMSE, Lehigh University, Bethlehem, Pennsylvania 18015

Abstract

We propose a planning model for multiple products manufactured across multiple manufacturing facilities sharing similar production capabilities. The need for a responsive product and capacity management framework is most evident in the fast-growing electronics and semiconductor industries. Our model is based on an emerging practice in these industries where product managers from varies business units dictate production planning across manufacturing facilities equipped with technologies to produce their products. We propose a multicommodity flow network model where each commodity represents a product manager's view and the network represents linked manufacturing facilities capable of producing the products. We proof that the single-commodity, multi-facility subproblem can be solved in polynomial time. We then develop a Lagragean decomposition, which separates the planning decisions into a number of product manager subproblems and a resource subproblem. Using subgradient search, these subproblems each propose solutions from its own perspectives while the Lagrangean multipliers penalize solution conflicts. The model structure and the solution methodology allow much flexibility for extensions in product and resource subproblems. We demonstrate that the base model can be solved efficiently with very small duality gaps.

1. Introduction

This research is motivated by production problems in the electronics, semiconductor and telecommunication industries. These industries struggle with their production planning problems in an increasingly complex supply chain structure. Specifically, to better utilize their capital-intensive equipment they are pressured to produce a wide variety of products in each of their production lines. However, these products may each belongs to a different supply chain operating under different delivery contracts, demand characteristics, and subcontracting agreements. As a result, detailed planning decisions are often relegated to product managers who are most familiar with their specific customer and supplier issues. On the other hand, resource consolidation and capacity management issues must be considered in a globally consistent fashion within each manufacturing facility.

Coordinating production under complex supply structure is not a new problem. However, two recent trends in these industries exacerbate the intensity of the problem. First, the trend toward increased market responsiveness intensifies the inter-dependency within the supply chain. In the past, excess inventory was generally used to reduce the impact of variation across different facilities. Today, most manufacturers are moving away from carrying substantial inventories, a trend that can be attributed to the widespread application of just-in-time manufacturing. Second, the rate of technological innovation significantly shortens the life span of manufacturing equipment, which in term increases the cost of manufacturing capacity. This combined with increased product variety and decreased product volumes prompt manufactures to cross-load their manufacturing facilities.

In this paper, we focus our study on the operational planning issues faced by the manufacturing facilities in this increasingly complex environment. The subject of supply chain management has attracted a lot of attentions recently. Quantitative analysis of supply chain issues are most commonly addressed using extensions of multi-echelon inventory models. Cohen and Lee (1988)(1989), Sterman (1989) and Davis (1993) are among the pioneers who made significant contribution in this area. Various development of these models is currently an area of active research (c.f., Lee, et al. 1995, Hahm and Yano 1995a,b and Arntzen, et al, 1995). A related line of research focuses on the extension of decision models in the traditional MRP systems (c.f. Billington, et al., 1983, Carlson and Yano, 1983, Gupta and Brennan 1995). This paper extends beyond the current literature in three important ways:

1. Whereas much of the supply chain literature focuses on the design and analysis of the entire supply chain, we focus our attention on the operations planning issues faced by manufacturing facilities sharing common production capabilities. Rather than decoupling multiple supply chains so that they can be treated in isolation we examine

a model which integrate decisions across multiple supply chains from the vintage point of manufacturing facilities.

2. Current literature emphasizes on applications in build-to-stock retail and distribution environments, this work attempts to mirror the practice in build-to-order manufacturing environments.
3. Most existing work ignore the trend that decision making in industrial environments are increasingly distributed, localized and product oriented. We propose a multicommodity flow model which has a build-in structure for product based decomposition in a multiple-facility environment.

From a modeling point of view our model is most closely related to the literature in multi-level, multi-period lot-sizing models with limited capacity. A number of survey articles (c.f., Bahl et al. 1987, Goyal and Gunasekaran, 1990; Baker 1993; Kimms 1997) provide an excellent review for research in this area. Our proposed model has two distinctive features that are not addressed in the existing literature. One is the explicit consideration of facility selection decisions. Most existing work assumes either a single facility, or multiple tiers of facilities as defined by the product structure, but no *facility selection* decisions are included. Second, we propose a solution method using the notion of Lagrangean Decomposition (Guignard and Kim, 1987) and variable splitting (Jörnsten 1986). This allows us to decompose the model into a resource subproblem and a number of product subproblems. The decomposition allows rich potentials for the model to be expanded to include product-specific considerations and facility-specific submodels. Lagrangean decomposition Similar approach to the single-facility case has been proposed by Thizy (1991) and Millar and Yang (1993). A key to the success of our solution method is that we proof the multi-facility, single-product subproblem with setup is solvable in polynomial time.

2. A Multi-Facility Production Model

We now consider a multi-facility production problem where a set of end items is to be produced in multiple facilities over multiple stages and multiple periods. Each end-item has a bill of material described by a *product structure*. There is an underlying *supply structure* where a set of alternative facilities could be setup to produce each item described in the product structure. Figure 1 contains an illustration of the product and the supply structure. The *product structure* in Figure 1 can be represented by a typical "gozinto" structure, widely adopted in the lot-sizing literature. The gozinto structure is specified an $n \times n$ matrix $[a_{ik}]$ where a_{ik} is the number of item i that is (directly) needed to produce one unit of item k. In additional to the gozinto structure we define the *supply structure matrix* $[r_{ij}]$ where $r_{ij} = 1$ if facility j could be used to produce item i, and $r_{ij} = 0$ otherwise.

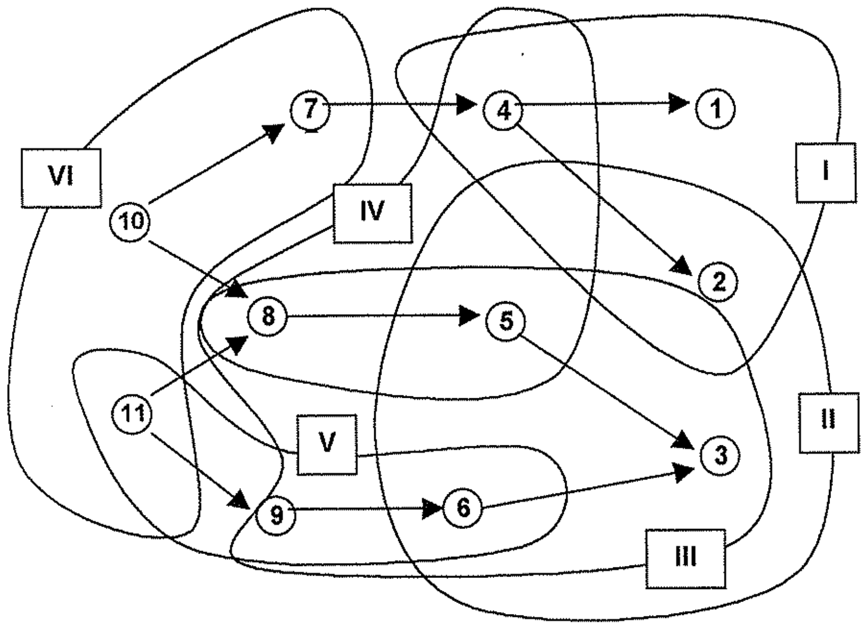


Figure 1. A Supply Network with Three End-Items, Six Facilities and a Maximum of Three Alternative Facilities for an Item

2.1 A Multicommodity Flow Model

The above multi-facility production problem is complex in that the facility selection decisions are combined with multi-item, multi-stage, multi-period production decisions. To approach this problem we first take the viewpoint of a subset of manufacturing facilities in the supply network. Each manufacturing facility is capable of producing a variety of products (items). We consider the production of multiple products (i=1,2,...,n) over multiple periods (t=1,2,..,T) where each item i can be produced using one of a specified set of alternative production facilities (j=1,2,.., J_i). Each facility can be setup to perform a limited number of production processes with a setup cost. Now consider a multicommodity flow network G(N,A) where each item i corresponds to a commodity in the network. Let $D_t^{\ i}$ denote the *internal demand* for item i in period t based on the end-item demand and the product structure. Suppose $D_t^{\ i}$ can be generated *a priori*, we can then define a multicommodity network as shown in Figure 2. This multicommodity flow network has three parts: a set of source nodes, a set of sink nodes, and a set of *production submodels* in between. Each commodity (product) i has a source node s^i , and T sink nodes d^i_t, one for each period t.

The input flow for source node s^i is the total demand over T periods for item i $(\sum_{t=1}^{4} D_t^i)$, and the outflow on sink note d^i_t as the demands for item i in period $t,(D_t^i)$. The arcs going from the source nodes s^i to the facility subnet j represent facilities that could be setup to produce item i. These arcs are specified by the *supply structure matrix*, i.e., there is an arc (i,j) for each non-zero entry of matrix $[r_{ij}]$. The subnetworks between the set of source and sink nodes represent production facilities shared by multiple items. The production resource subnetwork is to be "customized" according the structure of the particular manufacturing facility. For example, the simple structure in Figure 2 shows the familiar multi-period dynamic lot sizing model which can be used to represent individual production lines. This can be easily extended to a multi-stage model (c.f., Afentakis 1984) also shown in the figure, or other multi-period models. A variety of production line models in the literature (c.f., Graves 1992) could be incorporated in this general framework. Throughout this paper, we use the dynamic lot-sizing model to demonstrate the general structure of the supply-chain planning model. The arcs going from facility subnet j period t to sink node d^i_t represent the fact that the demand of item i in period t can be satisfied by the production and/or the inventory from all of its alternative facilities.

Each arc in the network is characterized by (f^i, c^i, u): arc flow f^i, per unit cost c^i and arc capacity u. The interpretation of these values varies according to the types of arcs. The arcs going from the source nodes s^i to the facility subnet j are *facility selection arcs* $A_s \subset A$, charaterized by (x^i_j, c^i_j, u_{ij}): x^i_j represents the total production to be performed on facility j over $t=1,..,T$, the arc cost c^i_j can be used to quantify differences among facilities (e.g., quality reputation), and capacity u_{ij} represents the maximum amount of item i that can be produced in facility j. When the dynamic lot sizing model is used as the *production submodel,* there are two types of arcs within the subnet: arcs going from left to right are *production arcs* $A_P \subset A$, characterized by $(x^i_{jt}, c^i_{jt}, cap_{jt})$: production volume x^i_{jt}, unit production cost c^i_{jt} ,and production capacity $cap_{jt} = u$. Arcs going from top down are *inventory arcs* $A_I \subset A$ characterized by $(I^i_{jt}, h^i_{jt}, inv_{jt})$: inventory carried from period t to t+1, I^i_{jt}, unit inventory holding cost c^i_{jt} and inventory limit $inv_{jt} = u$. Finally, the arcs going from facility j period t to sink node d^i_t are *demand arcs* $A_d \subset A$, *charcterized by* $(b^i_{jt}, c^i_{jt}, cap_{jt})$: facility j's contribution to demand D^i_t, b^i_{jt}, ,transportation cost c^i_{jt} ,and transportation capacity in period t, cap_{jt}.

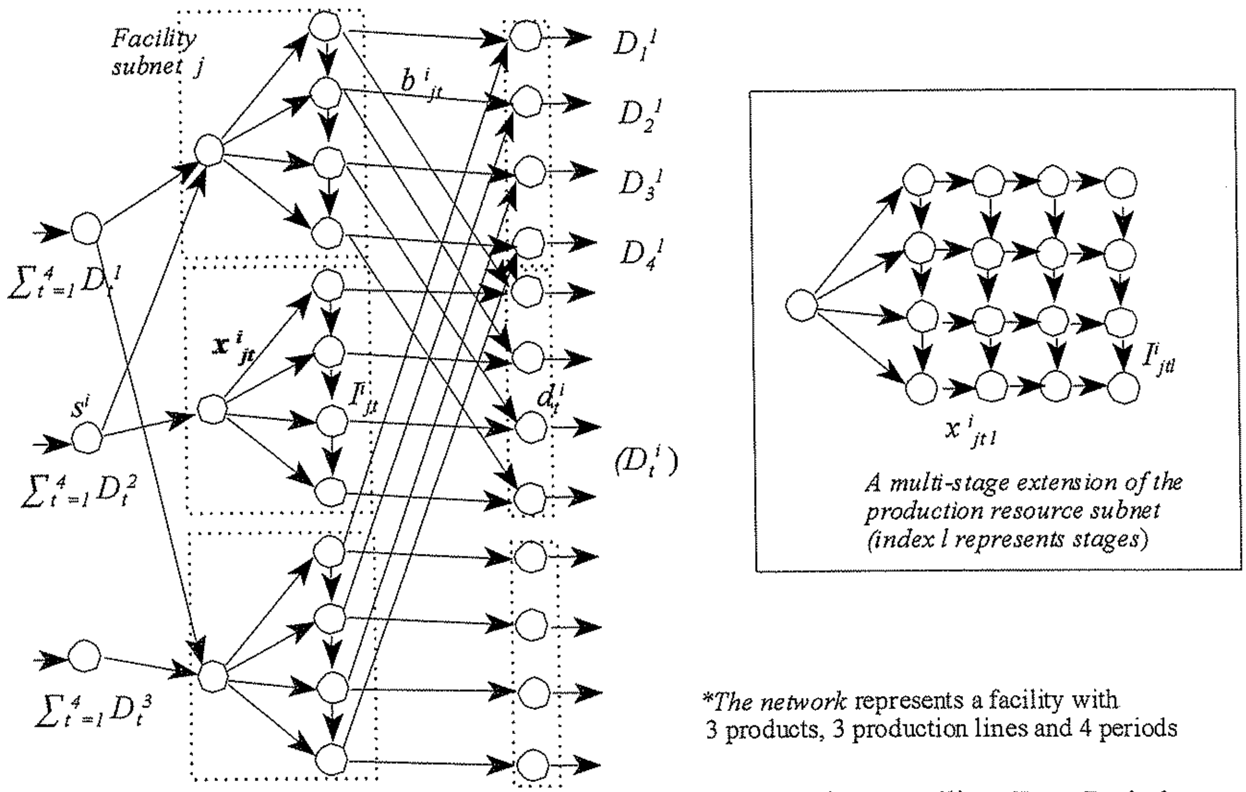


Figure 2. A Multicommodity Flow Network for a Three-Item, Three-Facility, Four-Period Model

Given the above specification, we can define the *general multicommondity flow constrains (1.1)* and *(1.2)* as follows:

General arc capacity constraints for all arcs

$$\begin{aligned} & x^i_j \le u_{ij} && \forall (i,j) \in A_s \\ & \sum_{i=1}^{n} \beta^i_{jt} f^i_{jt} \le u_{jt} && \forall (j,t) \in A_p \cup A_I \cup A_d \end{aligned} \qquad (1.1)$$

Denote $\mathcal{N}$ the node-arc incidence matrix for the multicommodity flow network G(N,A) and e^i the net balance flows for commodity i. The mass balance constraints are as follows:

Mass balance constraints for each commodity

$$\mathcal{N} f^i = e^i \qquad \forall i \in N \qquad (1.2)$$

In addition to the general multicommodity flow constraints as specified by the network structure G(N,A), we may need to define additional constraints for each facility submodel. Consider the multi-period lot-sizing model we use for all facility subnets. The inventory balance constraints are already included in the mass balance constraints (1.2). However, we will need to define additional constraints due to setup. Let α^i_{jt} denote the rate of capacity consumption for setup activities, q^i_{jt} a binary variable indicating the existence of a setup for item i at facility j at period t. The *production specific constraints* are as follows:

Production capacity constraints for production and setup

$$\sum_{i=1}^{n}\sum_{j=1}^{J_i}(\beta^i_{jt}x^i_{jt} + \alpha^i_{jt}\delta^i_{jt}) \le cap_{jt} \qquad \forall (j,t) \in A_p \qquad (2.1)$$

Setup constraints

$$x^i_{jt} \le Mq^i_{jt} \qquad \forall i \in N, (j,t) \in A_p \qquad (2.2)$$

Up to this point we ignore the fact that the items entering the production facilities (as represented by the multicommodity flow network G(N,A)) has an underlying product structure $[a_{ik}]$. To model the *supply structure* as well as the *product structure* across multiple tiers of a supply chain, we will need to include additional constraints which specify the relationship among demands of items over time D^i_t, $i=1,\dots,n$, $t=1,\dots,T$. First of all, the demand for end-items, $N_0 \subseteq N$, must be satisfy. Although this is already implied in the mass balance constraints (1.2), we will restate it for clarity.

Demand for end item must be satisfied

$$\sum_{j=1}^{m} b^i_{jt} = D^i_t \qquad \forall t, i \in N_0 \qquad (3.1)$$

The end-item demand triggers the internal demands in the supply chain as defined by the product structure. Denote L_k the lead-time for item k, we can define the following relationship:

Demand internal the supply chain must be satisfied in each period

$$\sum_{j=1}^{m} b^i_{jt} = \sum_{k=1}^{n} a_{ik} D^k_{t+L_k} \qquad \forall t, i \in N - N_0 \qquad (3.2)$$

Denote c^i_{jt} and h^i_{jt}, respectively, the production and the inventory holding costs for item i at facility j during period t, and K^i_j, the set-up cost for item i at facility j. We define a linear objective function with two main components: (1) a cost function $S_i(f^i, c^i, u)$ defined based on the supply chain requirements specific to item i, and (2) major cost components defined based on the production submodel. A linear multicommondity flow formulation of the multi-facility production problem **(P)** is given by:

Problem (P):

$$\textit{Minimize} \;\; z = \sum_{i=1}^{N} S_i(f^i_{jt}, c^i_{jt}, u_{jt}) + \sum_{t=1}^{T}\sum_{i=1}^{n}\sum_{j=1}^{J_i}(c^i_{jt}x^i_{jt} + K^i_j\delta^i_{jt} + h^i_{jt}I^i_{jt})$$

s.t.

<General Multicommodity Flow Constraints (1.1)-(1.2)>

<Production Specific constraints (2.1)-(2.2)>

<Supply Chain Specific Constratins (3.1)-(3.2)>

(4) nonnegativity constraints

$$f^i, x^i_{jt}, \delta^i_{jt}, I^i_{jt}, b^i_{jt} \geq 0$$

(5) binary constraints

$$\delta^i_{jt} \in (0,1)$$

It is useful to note that in this multicommodity flow model, only the arc capacity constraints (1.1) and (2.1) are bundling constraints. All the other constraints can be decomposed by commodity. (P) is a multi-period, multi-item, multi-facility production planning model. The base model can be expanded in two directions: (1) incorporate product-specific supply chain constraints in each commodity subproblem, and (2) incorporate facility-specific production submodels and focus on facility selection decisions. The former is the focus of a related paper by (Meixell and Wu, 1998). In this paper, we focus on the latter facility selection problem. In the remainder of the paper, we explore several special structures of this problem and propose a solution methodology.

2.2 Model Analysis

Note that since the mass balance constraints (1.2) imply $\sum_{j=1}^{m} b^i_{jt} = D^i_t, \forall i \in N$, constraints (3.2) in effect specify the relationship between the demands of item *i* and the items (*k's)* which use *i* as a component, i.e.,

$$D^i_t = \sum_{k=1}^{n} a_{ik} D^k_{t+L_k}, \forall t, i \in N - N_0 \qquad (3.3)$$

Consider a set of facilities in a manufacturing supply chain as depicted in Figures 2. Suppose the end-item demand is stationary, the relationship in (3.3) suggest that it is possible to generate D^i_t *a priori* using a BOM explosion mechanism frequently used in MRP systems, i.e., we may treat D^i as parameters generated and fixed *a priori* for the multicommodity flow model such that (3.2) is always satisfied. This simplifies the facility selection decisions considerably since we do not need to make multi-tier production planning decisions at the same time. The two decisions are often considered separately in practice since demands across different tiers of the supply chain are decoupled by built-in leadtime and inventory buffers. For the product-focused extension of the base model (Meixell and Wu, 1998), the multi-tier production planning problem is examined in greater detail.

As stated above the model *(P)* can be decomposed by commodity after relaxing the bundling capacity constraints. Different decompositions of the multicommodity flow problems are well documented in the literature (c.f., Assad (1980), Kennington and Helgason (1980)). Consider for a moment the capacity-relaxed single-item, multiple-facility subproblem for commodity *i* as follows:

$$(P_i) \quad Minimize\ S(f_{jt}, c_{jt}, u_{jt}) + \sum_{t=1}^{T}\sum_{j=1}^{J_i}(c_{jt}x_{jt} + K_j\delta_j + h_{jt}I_{jt})$$

s.t. <*mass balance constraints for commodity i (1.2)*>
<*setup constraint for commodity i (2.2)*>
<*constraints (4)(5) for commodity i*>

This problem corresponds to a decision problem for the product manager of item *i*, which could be produced on a particular set of manufacturing facility J_i. Without loose of generality, we may incorporate in the submodel a performance measure $S(f_{jt}, c_{jt}, u_{jt})$ specific to item *i* according the particular requirements for product *i*. This performance measure can be due-date based, measured against a shipment (e.g., truck) schedule, or justified according to other product specific measures.

The *single* facility, single item uncapacitated lot-sizing problem has been studied intensively in the literature. Despite of its binary variable it is well known that this problem can be solved in polynomial time using Wagner-Within type algorithms. In recent years, more efficient implementation of Wagner-Within algorithms has been developed by Federgruen and Tzur (1991) and Wagelmans et al. (1992), which has order $O(n \log n)$ or better. The existence of these polynomial time solvable subproblems allow the more general multi-period, multi-item capacitated lot-sizing problems to be solvable in a reasonable amount of time for realistic size problems (c.f. Tempelmeir and Derstroff, 1996). Since our primary concern here is the *multiple* facility case we are interested to know if the single-item, multi-facility uncapacitated subproblem can be solved in polynomial time. There are two important reasons for this: (1) the subproblem has the form of a mixed integer program. Unless special structures exist, the solution to problem *(P)* is unlikely to be efficient, and (2) there is no straightforward (efficient) decomposition from the multi-facility case to single-facility. We now present a few important result for the multi-facility subproblem.

Theorem 1. ***(Non-splitting property):*** *There exists an optimal solution to the single-item, multiple facility problem (P_i) such that item i's demand in period t is produced in exactly one of the J_i facility, i.e., exactly one of the b^i_{jt} is positive for each period t.*

Proof: It is easy to verify that problem *(P_i)* has Leontief structures. The setup cost K^i_j can be incorporated into the production cost C^i_{jt} as a fixed charge function as below:

$$C^i_{jt} = \begin{matrix} c^i_{jt}x^i_{jt} + K^i_j & \text{if } x^i_{jt} > 0 \\ 0 & \text{if } x^i_{jt} = 0 \end{matrix}$$

Other constraints are linear while the objective function is concave. Thus model *(P_i)* has the following features: all nonnegative variables x, I, b, appear exactly once with a positive (+1) coefficient; in all other occurrences they have a negative (-1) coefficient. It follows that if more than one variable appears with a positive coefficient in the same constraints, then only one of these variables can be positive in the optimal solution, which results in the following conditions:

$$b^i_{jt}b^i_{kt} = 0 \quad \text{for } t=1..T, j=1..J_i, k=1..J_i,$$

This condition states the nonsplitting property. □

Theorem 2. *There exists an optimal solution to the single-item, multiple facility problem (P_i) that has the following properties:*

(i) *No simultaneous production of item i over more than one facility can take place in a given period. In other words,* $x^i_{jt}x^i_{kt}=0, \ \forall i,j,k,t.$

(ii) *No production of item i will be scheduled at all if there is inventory carried over from a previous period in one of the facilities. In other words,* $x^i_{jt}I^i_{kt}=0, \ \forall i,j,k,t.$

Proof: Consider the following mass balance constraints implied from (1.2):

$$x^i_{jt} + I^i_{jt-1} = b^i_{jt} + I^i_{jt} \qquad \forall i \in N, j \in J_i, t \in T$$

$$\sum_{j \in J} b^i_{jt} = D^i_t \qquad \forall i \in N, t \in T$$

These constrains can be replaced by the following constraint:

$$\sum_{j=1}^{J_i} (x^i_{jt} + I^i_{jt} - I^i_{jt}) = D^i_t \qquad \forall i \in N, t \in T$$

With this new constraint problem *(P_i)* still has Leonteif structures which implies the following conditions:

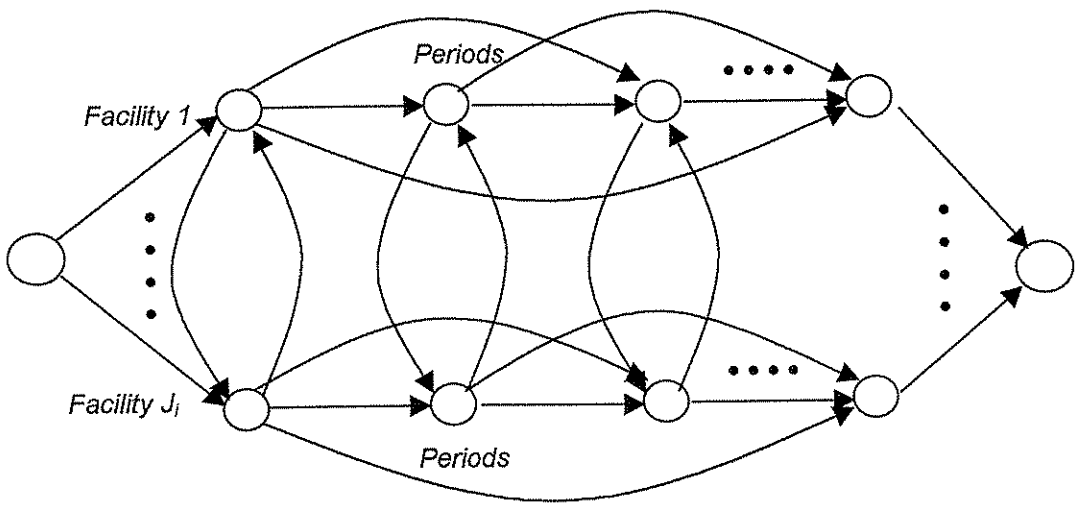
$$x^i_{jt}x^i_{kt} = 0 \quad \text{for } t=1..T, j=1..J_i, k=1..J_i,$$

$$x^i_{jt}.I^i_{kt-1} = 0 \quad \text{for } t=1..T, j=1...J_i, k=1..J_i, \qquad □$$

Note that the conditions stipulated in Theorem 2 is more restrictive than that of Theorem 1. It states that there exists an optimal solution where not only an item's *demand is satisfied* by exactly one facility in a given period (non-splitting), but no production will be scheduled for the item in more than one facility during any given period. Furthermore, if there is inventory exist in some facility from a previous period (that is available at a lower cost), there will be no production scheduled for the current period at all. Theorems 1 and 2 lead to an important result as follows.

Theorem 3. *The uncapacitated, single-item, multiple-facility problem can be solved in polynomial time using a shortest path algorithm.*

Proof: We will state the proof using a familiar graphical representation as follows.



The first row of nodes denotes facility 1 and the *ith* row denote facility J_i. There are $T+1$ time epochs:0,1,2,3, and a period is the interval between epochs, i.e., between epochs 0 and 1 is period $t=1$, and between 1 and 2 is period $t=2$, etc. A horizontal arc denotes the production that satisfies all the periods' demand within the time epochs. The arc cost includes production, inventory and setup costs. A vertical arc denotes a switch from one facility to another and the cost associated to these arcs are 0. There is an artificial source and sink, arcs adjacent to these nodes have 0 cost. In a general graph for each time epoch there are arcs for all the facility periods. So production of an item may switch from one facility to any other in different periods. From Theorem 2, there will be production scheduled for item i only if there is no inventory carried over from a previous period in one of the facilities. In other words, in an optimal solution exactly one arc will be chosen to enter a given node in the network. On the other hand, Theorem 2 states that there can be no simultaneous production of items i in more than one facility in any given period. In other words, in an optimal solution exactly one arc will be chosen to leave a node in the network. This means that an optimal production schedule corresponds to a source-to-sink path in the network, and it corresponds to a shortest cost path. □

The special structure of problem (P_i) suggest an optimization algorithm for problem (P) which relax the capacity constraints and solves repeatedly the uncapacitated problem (P_i) for increasingly better bounds. In the following, we describe a solution methodology making use of the special structure.

3. Solution Methodology

We propose a Lagrangean Decomposition scheme for the solution of problem (P). Lagrangean Decomposition was first proposed by Guignard and Kim (1987). The scheme has been applied to a variety of NP-hard problems including multi-item single-facility lot-sizing problems (Thizy, 1991). A main advantage of Lagrangean Decomposition over the better known Lagreangean Relaxation is that the theoretical Lower Bound obtained from Lagrangean Decomposition at least as tight as that from Lagrangean Relaxation. We start our exposition by first listing the mass balance constraints (1.2) explicitly:

$$\sum_{t=1}^{T} D_t^i = \sum_{j=1}^{J_i} f_j^i \qquad \forall i \in N \qquad (1.2.1)$$

$$f_j^i = \sum_{t=1}^{T} x_{jt}^i \qquad \forall i \in N, j \in J_i \qquad (1.2.2)$$

$$x_{jt}^i + I_{jt-1}^i = b_{jt}^i + I_{jt}^i \qquad \forall i \in N, (j,t) \in A \qquad (1.2.3)$$

$$\sum_{j=1}^{J_i} b_{jt}^i = D_t^i \qquad \forall i \in N, t \in T \qquad (1.2.4)$$

If we assume that the system must return to its initial inventory at the end of the planning horizon, i.e., $I_0 = I_T$, we may simplify the above mass balance constraints. From (1.2.3) and (1.2.4) we have,

$$D_t^i = \sum_{j=1}^{J_i} (x_{jt}^i + I_{jt-1}^i - I_{jt}^i), \forall i,t, \text{ thus } \sum_{t=1}^{T} D_t^i = \sum_{t=1}^{T}\sum_{j=1}^{J_i} (x_{jt}^i + I_{jt-1}^i - I_{jt}^i) \qquad (1.2.5)$$

but since $I_0 = I_T$

$$\sum_{t=1}^{T} I_{jt-1}^i = I_0 + \sum_{t=1}^{T-1} I_{jt}^i = \sum_{t=1}^{T-1} I_{jt}^i + I_T = \sum_{t=1}^{T} I_{jt}^i, \text{ so we can rewrite (1.2.5) as}$$

$$\sum_{t=1}^{T} D_t^i = \sum_{j=1}^{J_i}\sum_{t=1}^{T} x_{jt}^i, \forall i \in N \qquad (1.2.6)$$

Note that (1.2.6) implies constraints (1.2.1) and (1.2.2). This allows us to consider problem (P) with only two sets of balance constraints (1.2.3) and (1.2.4) since (1.2.1) and (1.2.2) will be satisfied automatically.

The basic idea of our decomposition is to separate the multicommodity flow problem *(P)* into two subproblems: one with the capacity and the mass balance constraints, but not the setup constraints, the other is a commodity-decomposable subproblem with the mass balance and the setup constratins. The second subproblem is separable to single-commodity problems *(P_i)*, which has special structure as stated in Theorems 1 to 3. In this decomposition, the first subproblem is a linear program and the second is a collection of shortest path problems, all are relatively easy to solve. To demonstrate this solution methodology we use a slightly simplified formulation of problem (P) by dropping the first term in the objective, and assuming that setup does not consume capacity (dropping (2.1)). We then restate the multi-facility production problem **(P')** as follows:

$$\text{Minimize} \quad z=\sum_{t=1}^{T}\sum_{i=1}^{n}\sum_{j=1}^{J_1}(c^i_{jt}x^i_{jt}+K^i_j q^i_{jt}+h^i_{jt}I^i_{jt})$$

$s.t.$

$\sum_{i=1}^{n}\beta^i_{jt}x^i_{jt}\le u_{jt}$	$\forall (j,t)\in A$	(1.1)
$x^i_{jt}+I^i_{jt-1}=b^i_{jt}+I^i_{jt}$	$\forall i\in N,(j,t)\in A$	(1.2.3)
$\sum_{j=1}^{J_1}b^i_{jt}=D^i_t$	$\forall i\in N,t\in T$	(1.2.4)
$x^i_{jt}=xx^i_{jt}$	$\forall i\in N,j\in J,t\in T$	(6.1)
$b^i_{jt}=bb^i_{jt}$	$\forall i\in N,j\in J,t\in T$	(6.2)
$I^i_{jt}=II^i_{jt}$	$\forall i\in N,j\in J,t\in T$	(6.3)
$xx^i_{jt}+II^i_{jt-1}=bb^i_{jt}+II^i_{jt}$	$\forall i\in N,j\in J,t\in T$	(1.2.3)'
$\sum_{j\in J}bb^i_{jt}=D^i_t$	$\forall i\in N,t\in T$	(1.2.4)'
$xx^i_{jt}\le Mq^i_{jt}$	$\forall i\in N,j\in J,t\in T$	(2.2)
$x^i_{jt},I^i_{jt},b^i_{jt}\ge 0$	$\forall i\in N,j\in J,t\in T$	(4.1)
$xx^i_{jt},II^i_{jt},bb^i_{jt},\delta^i_{jt}\ge 0$	$\forall i\in N,j\in J,t\in T$	(4.2)
$\delta^i_{jt}\in(0,1)$	$\forall i\in N,j\in J,t\in T$	(5)

In the above formulation, we make copies of the variable x^i_{jt}, b^i_{jt}, and I^i_{jt} as xx^i_{jt}, bb^i_{jt}, and II^i_{jt}. We then use the copies to split the original constraints into two sets of constraints: {(1.1),(1.2.3),(1.2.4),(4.1)} and {(1.2.3)',(1.2.4)',(2.2),(4.2),(5)} plus the linking constrains (6.1)-(6.3). We then separate **(P')** by relaxing these linking constraints and placing them in the objective function with Lagrangean multipliers λ^x_{ijt}, λ^I_{ijt}, and λ^b_{ijt}. This yields the following subproblems:

Resource Subproblem:

$$\text{Minimize} \quad z_1=\sum_{t=1}^{T}\sum_{i=1}^{n}\sum_{j=1}^{J_1}((c^i_{jt}+\lambda^x_{ijt})x^i_{jt}+(h^i_{jt}+\lambda^I_{ijt})I^i_{jt}+\lambda^b_{ijt}b^i_{jt})$$

$s.t.$ (1.1),(1.2.3),(1.2.4),(4.1)

***n* Product Subproblems:**

Minimize $$z_2 = \sum_{i=1}^{n}(\sum_{t=1}^{T}\sum_{j=1}^{J_i}(-\lambda_{ijt}^{x}xx_{jt}^{i} - \lambda_{ijt}^{I}II_{jt}^{i} - \lambda_{ijt}^{b}bb_{jt}^{i} + K_j^i q_{jt}^i)) = \sum_{i=1}^{n} z_2^i(\lambda)$$

$s.t.$ (1.2.3)′, (1.2.4)′, (2.2), (4.2), (5)

Note that under the general framework of Lagrangean Decomposition, constraints (1.2.3) and (1.2.4) do not need to be duplicated for *both* subproblems, i.e., these constraints can be assigned to *either* subproblems. However, constraint duplication does improve the speed of convergence since the solutions proposed by the subproblems tend to be more similar. Our computational experience indicates that as long as the added constraints do not add computational burden to the subproblems, duplication improves solution performance. A lower bound to problem **(P′)** given Largrangean multiplier set λ is as follows:

$$LB_\lambda(P') = v(z_1(\lambda)) + \sum_{i=1}^{n} v(z_2^i(\lambda))$$

Where $v(.)$ denote the optimal value of the problem. Note that the resource subproblem is a linear program, and the product subproblem z^i_2 has identical structure as problem (P_i) described earlier, which can be solved efficiently using a shortest path algorithm. Given the solutions of z_1 and z_2 one could find an upper bound using the following feasibility restoration routine: given the solution for the resource subproblem we add setups for the periods where production is nonzero, i.e., we set q^i_{jt} to 1 whenever $x^i_{jt}>0$. Then we calculate the objective function using original cost function. This results in an upper bound for the original problem. The lower bound can be maximized by searching for the set of Largrangean multipliers λ that maximize the Lagrangean dual. Both dual ascent and subgradient search methods can be used for this task. In this paper, we use the later approach, which is summarized in Section 3.2. As we will demonstrate in the computational section, we can achieve solutions with very small duality gaps using the bounds and the search algorithm.

3.1. Managerial Insights Related to the Decomposition

Our choice of Lagrangean Decomposition is not purely motivated by computing. The decomposition of the multi-facility production model into a resource subproblem and multiple product subproblems has interesting managerial implications. As recognized by several researchers (c.f., Jörnsten and Leisten, 1994; Burton and Obel, 1984), mathematical decomposition often leads to insights for general modeling strategies or even new decision structures. The decomposition suggested earlier allows further analysis concerning modeling flexibility in the context of multi-facility manufacturing planning. Suppose we consider each product subproblem as a decision problem for a *product* manager and the resource subproblem as a decision problem for a *production* manager overseeing multiple facilities. Thus, the decomposition can be viewed as a decision system where *product* managers, each responsible for a product, compete for resource capacity available from manufacturing facilities. The *production* manager, on the other hand, represents the interests of efficiently allocating resources from multi-facilities to the products. Clearly the solutions proposed by the production manager (x, I, b) do not agree with the collective solution proposed by the product managers (xx, II, bb). A search based on Lagrangean multipliers essentially penalizes their differences, while adjusting the penalty vector iteratively. This process stops when the degree of disagreement (the duality gap) is acceptably low, or when further improvement is unlikely.

The above viewpoint is useful in evaluating the flexibility model *(P)* represents. First, it should be clear that each product subproblem (P_i) could be customized to represent the distinctive needs of each product. So long as its basic network structure is maintained there will be no additional computational burden. Similarly, as long as the resource subproblem remains a linear program, it can be customized with various facility submodels each reflecting the distinct production structure of a facility. However, a different constraint duplication strategy may be necessary when changes are made to the base model.

3.2. The Subgradient Search Algorithm

In this section, we summary the subgradient search algorithm used to adjust the Lagrangean multipliers. At each iteration s, we calculate Lagrangean multipliers using the following equations:

$$\begin{aligned} \lambda_{ijt}^{x,s+1} &= \lambda_{ijt}^{x,s} + u^s (x_{jt}^{i,s} - xx_{jt}^{i,s}) \\ \lambda_{ijt}^{I,s+1} &= \lambda_{ijt}^{I,s} + u^s (I_{jt}^{i,s} - II_{jt}^{i,s}) \\ \lambda_{ijt}^{b,s+1} &= \lambda_{ijt}^{b,s} + u^s (b_{jt}^{i,s} - bb_{jt}^{i,s}) \end{aligned} \qquad (7)$$

where

$$u^s = \frac{\gamma_s (UB_s^* - (v(z_1(\lambda^s)) + \sum_{i=1}^{n} v(z_2^i(\lambda^s)))}{\sum_{t=1}^{T}\sum_{i=1}^{n}\sum_{j=1}^{J_i} ((x_{jt}^{i,s} - xx_{jt}^{i,s})^2 + (I_{jt}^{i,s} - II_{jt}^{i,s})^2 + (b_{jt}^{i,s} - bb_{jt}^{i,s})^2)} \qquad (8)$$

γ_s: a scalar set to 1 and reduced by half if the lower bound fails to improve after a fixed number of iterations

UB^*_s: the best upper bound obtained up to iteration s

We terminate the algorithm after a prespecified number of iterations. The best upper bound obtained at the end of the iterations is the heuristic solution to the problem. We summary the algorithmic steps are as follows:

Step 1: Initialize s, λ, u, γ and UB^*.

Step 2: Solve the resource and the product subproblems. Compute the lower bound $(v(z_1(\lambda^s)) + \sum_{i=1}^{n} v(z_2^i(\lambda^s))$ for current iteration, s.

Step 3: Compute an upper bound UB_s from the optimal solution of the current resource subproblem $Min\ z_1(\lambda^s)$. If $UB_s < UB^*_{s-1}$, set $UB^*_s \leftarrow UB_s$.

Step 4: Update the multipliers using equations (7) and (8)

Step 5: Stop if a prespecified iteration limit is reached. Otherwise go to Step 2.

4. Computational Testing

We coded the subgradient search algorithm using the mathematical programming language AMPL along with the CPLEX linear solver. The experiments are conducted on a Pentium-200 personal computer with 64Meg RAM. We first generate a set of 10 benchmark test problems with 100 products, 5 facilities and 5 periods. We then alter the problem characteristics and sizes to generate eight additional test sets each with 10 instances. For the benchmark problems the production, inventory and setup costs are randomly generated using Uniform distribution. Demands are also randomly generated using Uniform distribution. To generate capacity we use the following procedure: we first calculate cumulative demands by adding the randomly generated demands of all items up to period t for all $t \in T$. For the first period, we multiply the total demand for the period by some constant (≥ 1). We then use this number as the total capacity available in the period and assign a fraction to each facility. For the coming periods, total capacities assigned for the previous time periods are subtracted from the cumulative demand of that period and then multiplied by some constant to generate the capacity. Using this procedure, we may generate relatively challenging (but feasible) test problems with tight capacity constraints. For the benchmark test problems theses constants are 1.6, 1.4, 1.3, 1.2, and 1.1 for periods 1 to 5, respectively.

For simplicity, we assume β^i_{jt} (the consumption rate of facility j's resoure by item i at period t) is equal to 1. We also assume the starting and ending inventory to be zero. In Table 1, we summarize the parameters used in the benchmark and each of the eight test sets (90 test problems total).

For most of the test problems lower bound increases significantly in the first 20 iterations whereas the upper bound improves slowly. There appears to be a strong correlation between the quality of the lower and the upper bounds, i.e., when the lower bound obtained is tight, the upper bound restored from the lower bound solution is also closer to optimum. We observed a quite consistent convergence pattern throughout all test problems. Figure 3 shows a representative convergence plot for the algorithm. As shown in the plot, convergence typically occurs quite early resulting in a very small duality gap.

For each test problem, we calculated the duality gap at the end of the iterations using the best upper and lower bound obtained during the search. The computational results in terms of the gap are summarized in Table 2. We summarize the observations as follows:

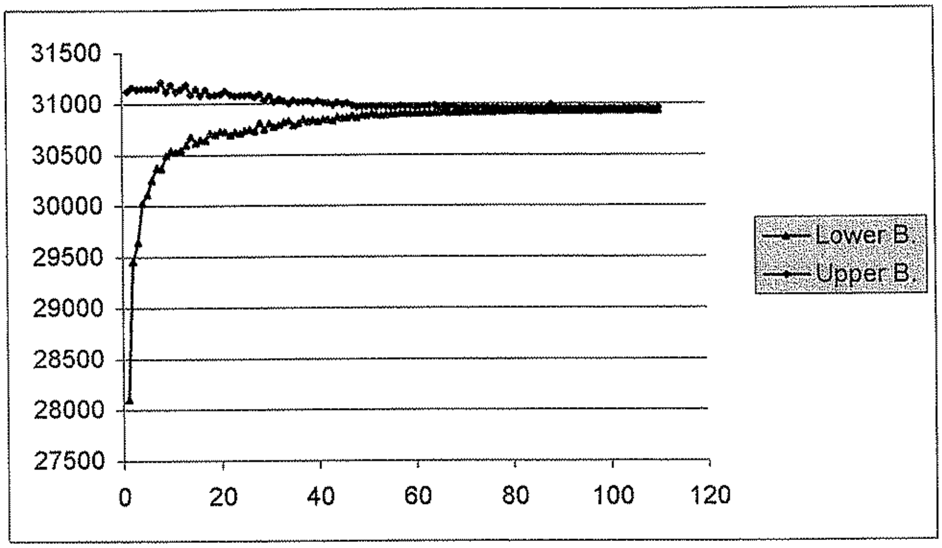


Figure 3. Representative Convergence Plot for the Subgradient Search Algorithm on a Benchmark Problem

1. As shown in the table, setup cost appears to have a significant effect on the duality gap. High setup instances has an average gap of 3.766% compared to 0.304% for the low setup instances. This result is not surprising since an increased setup costs widen the gap between the resource subproblem (which is an LP ignoring the setup cost) and the product subproblems. On the other hand, since the original problem is a mixed integer program with binary setup variables, as the setup costs increase the problem behave closer to a combinatorial problem then an LP.
2. The number of facilities seems to have an effect on the duality gap as well. The ten facility instances have a consistently higher gap (averaged 2.334%) when compared to the five facility benchmark (1.099%) and the one facility case (0.169%). This result is useful in that making multiple facility decisions is an unique feature of our

model. The results suggest that the added dimension has a noticeable effect on the difficulty of the problem. On the other hand, it also shows that the proposed algorithm is quite effective in solving the traditional single-facility problems.

3. The effect of capacity levels is much less pronounced. This may be due to the fact that the capacity generation procedure relatively tight capacity in all instances. Since the difference between non-capacitated and capacitated lot sizing models is well known, we did not make an attempt to further loosen the capacity.
4. Increasing the number of items does not seem to have the same effect as increasing the number of facilities. We tested four 300-item problems using exactly the same setting as the benchmark test set. The duality gaps are roughly the same as those in the benchmark problems, with an average of 0.967%. However, when we increase the number of items to 500, memory management becomes an issue.

Acknowledgements

This research is supported, in part, by National Science Foundation Grant DMI 9634808 and by the Microelectronics division of Lucent Technologies.

References:

Afentakis, P., B. Gavish and U. Karmarker (1984), "Computational Efficient Optimal Solutions to the Lot-Sizing Problems in Multistage Assembly Systems," *Management Science*, 30, 222-39.

Arntzen, Bruce C, Gerald G. Brown, Terry P. Harrison, and Linda L. Trafton (1995) "Global Supply Chain Management at Digital Equipment Corporation." *Interfaces* 25: 1 January-February 1995, pp. 69-93.

Assad, A.A., (1980) "Models for Rail Transportation," *Transportation Research*, 14A, 205-220.

Baker, K.R. (1993) "Requirements Planning" in *Logistics of Production and Inventory, Handbooks in Operations Research and Management Science, Volume 4*, ed. S.C. Graves, et al, Elsevier Science Publishers, Amsterdam, The Netherlands.

Bahl, H.C., Ritzman, L.P., and Gupta, J.N.D. (1987), "Determining Lot Sizes and Resource Requirements: A Review," *Operations Research*, Vol. 35, pp. 329-345.

Billington, Peter J., John O. McClain, and L. Joseph Thomas. (1983). "Mathematical Programming Approaches to Capacity-Constrained MRP Systems: Review, Formulation and Problem Reduction." *Management Science*, vol. 29, no. 10, October, pp 1126-1141.

Blackburn, J., and R. Millen (1982). "Improved Heuristics for Multi-Stage Requirements Planning Systems." *Management Science*, vol. 28, no. 1, pp. 44-56.

Burton, R.M., and Obel, B., (1984), *Designing Efficient Organizations*, North Holland, Amsterdam.

Carlson, Robert C., and Candace A. Yano (1983) "Safety Stocks in MRP - Systems with Emergency Setups for Components" *Management Science*, vol. 32, no. 4, April, pp 403-412.

Cohen, Morris A., and Hau L. Lee. (1988) "Strategic Analysis of Integrated Production-Distribution Systems: Models and Methods." *Operations Research*, vol. 36, no. 2, pp 216-228.

______ and ______ (1989) "Resource Deployment Analysis of Global Manufacturing and Distribution Networks." *Journal of Manufacturing and Operations Management*, no. 2, pp 81-104.

Davis, Tom. (1993) "Effective Supply Chain Management." *Sloan Management Review*, Summer, pp 35-46.

Federgruen, A. and M. Tzur (1981), "A Simple Forward Algorithm to Solve General Dynamic Lot Sizing Models with n Periods in *O(n long n)* or *O(n)* Time," *Management Science, 27, 1-18.*

Goyal, S.K. and Gunasekaran, A., (1990), "Multi-Stage Production-Inventory Systems," *European Journal of Operational Research*, Vol 46, pp. 1-20.

Graves, S.C., et al., (eds.) (1992) *Handbooks in Operations Research and Management Science, Volume 4, Logistics of Production and Inventory.* Amsterdam; Elsevier Science Publishers.

Guignard, M. and Kim, S. (1987), "Lagrangean Decomposition: A Model Yielding Stronger Lagrangean Bounds," *Mathematical Programming*, Vol. 39, pp. 215-228.

Gupta, S.M. and L Brennan. (1995) "MRP systems under supply and process uncertainty in an integrated shop floor control environment." *International Journal of Production Research*, vol. 33, no. 1, pp 205-220.

Hahm, Juho, and Candace Arai Yano, (1995a)"The Economic Lot and Delivery Scheduling Problem: the Common Cycle Case." *IIE Transactions* 27, 113-125.

______ and _____ (1995b), "The Economic Lot and Delivery Scheduling Problem: Models for Nested Schedules." *IIE Transactions* 27, 126-139.

Jörnsten, K.O., Näsberg, M. and Smeds, P.A. (1986), "Variable Splitting: A New Lagrangean Relaxation Approach to Some Mathematical Programming Models," Technical Report LiTH-MAT-R-85-04, Department of Mathematics, Linköping Institute of Technology, Sweden.

Jörnsten, K.O., and Leistern, R. (1994), "Aggregation and Decomposition for Multi-Divistional Linear Programs," *European Journal of Operational Research,* Vol. *72, 175-191.*

Kennington, J.L. and R.V. Helgason (1980), *Algorithms for Network Programming,* Wiley-Interscience, New York.

Kimms, A. (1997), *Multi-Level Lot Sizing and Scheudling,* Physica-Verlag, Heidelberg, Germany.

Lee, Hau, P. Padmanabhan, Seungjin Whang (1995), "Information Distortion in a Supply Chain: The Bullwhip Effect." Working Paper, Stanford University, April, 1995.

Meixell, M.J. and Wu, S.D. (1998), "A Modeling Framework for Demand Analysis in Manufacturing Supply Chains," *Technical Report,* Department of IMSE, Lehigh University.

Millar, H.H. and Yang, M. (1993), "An Application of Lagrangean Decomposition to the Capacitated Multi-Item Lot Sizing Problem," *Computers Ops. Res.*, Vol. 20, pp. 409-420.

Sterman, John D. (1989) "Modeling Managerial Behavior: Misperceptions of Feedback in a Dynamic Decision Making Experiment." *Management Science*, Vol. 35, no. 3, March.

Tempelmeir, H. and M. Derstroff (1996), "A Lagrangean Based Heuristic for Dynamic Multilevel Multiitem Constrained Lotsizing with Setup Times," *Management Science,* Vol 42, pp. 738-757.

Thizy, J.M., (1991), "Analysis of Lagrangian Decomposition for the Multi-Item Capacitated Lot-Sizing Problem," *INFOR*, Vol. 29, pp. 271-283.

Wagelmans, A., S. Van Hoesel and A. Kolen (1992), "Economic Lot Sizing: An *O(n log n)* Algorithm that Runs in Linear Time in the Wagner-Within Case," *Operations Research,* Vol 40, 145-155.

Table 1. Parameter Settings of the Test Set

Test Set #	1	2	3	4	5
	Benchmark	**Tight Capacity**	**Loose Capacity**	**High Holding Costs**	**Low Holding Cost**
Number of items	100	100	100	100	100
Number of facilities	5	5	5	5	5
Number of periods	5	5	5	5	5
Production cost	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[5,14]
Inventory holding cost	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[20,29]	Uniform ~[1,5]
Setup cost	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[5,14]
Demand	Uniform ~[1,10]	Uniform ~[5,14]	Uniform ~[1,10]	Uniform ~[1,10]	Uniform ~[1,10]
Capacity constants	1.6, 1.4, 1.3, 1.2, 1.1	1.3, 1.2, 1.1, 1.1, 1.05	1.8, 1.6, 1.5, 1.4, 1.3	1.6, 1.4, 1.3, 1.2, 1.1	1.6, 1.4, 1.3, 1.2, 1.1
Number of test problems	10	10	10	10	10
Number of iterations	106	106	106	106	106
Naming	BCH	TC	LC	HH	LH

Table 1b.

Test Set #	6	7	8	9
	High Setup Costs	**Low Setup Costs**	**10 Facilities**	**1 Facility**
Number of items	100	100	100	100
Number of facilities	5	5	10	1
Number of periods	5	5	5	5
Production cost	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[5,14]
Inventory holding cost	Uniform ~[5,14]	Uniform ~[5,14]	Uniform ~[20,29]	Uniform ~[1,5]
Setup cost	Uniform ~[20,29]	Uniform ~[1,5]	Uniform ~[5,14]	Uniform ~[5,14]
Demand	Uniform ~[1,10]	Uniform ~[1,10]	Uniform ~[1,10]	Uniform ~[1,10]
Capacity constants	1.6, 1.4, 1.3, 1.2, 1.1	1.6, 1.4, 1.3, 1.2, 1.1	1.6, 1.4, 1.3, 1.2, 1.1	1.6, 1.4, 1.3, 1.2, 1.1
Number of test problems	10	10	10	10
Number of iterations	106	106	108	110
Naming	HS	LS	10F	1F

Table 2. Duality gap[a] (in %) by test problems

	BCH	**TC**	**LC**	**HH**	**LH**	**HS**	**LS**	**10F**	**1F**
1	1.18	1.01	1.19	1.00	1.59	4.96	0.36	2.58	0.04
2	0.99	1.01	1.42	0.87	1.29	2.72	0.28	3.10	0.05
3	1.05	1.32	1.00	1.25	1.34	2.01	0.26	2.31	0.10
4	1.32	1.33	1.76	0.96	1.02	4.82	0.26	2.45	0.06
5	1.24	1.29	0.93	0.85	0.97	5.10	0.33	2.08	0.10
6	1.10	1.28	0.86	1.00	1.00	3.21	0.27	1.94	1.10
7	1.02	1.16	1.09	0.73	0.86	5.51	0.26	1.98	0.11
8	1.16	1.59	1.04	0.80	1.03	3.56	0.32	2.31	0.04
9	0.84	1.05	1.19	0.74	1.03	2.72	0.36	2.66	0.04
10	1.09	1.50	1.18	0.82	1.11	3.05	0.34	1.93	0.05
Avg.	1.099	1.254	1.166	0.902	1.124	3.766	0.304	2.334	0.169

[a] Duality Gap = $(\frac{UB^* - LB}{UB^*}) \cdot 100\%$