OPERATIONS RESEARCH CENTER

working paper

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

A Strong Cutting Plane Algorithm for Production Scheduling with Changeover Costs

by

Thomas L. Magnanti and Rita Vachani

OR 173-87 December 1987

A Strong Cutting Plane Algorithm
for Production Scheduling with Changeover Costs

T. L. Magnanti
Sloan School of Management, MIT

and

R. Vachani
GTE Laboratories

December 1987

* Research support in part by Grant ECS-8316224 from the Systems Theory and Operations Research Program of the National Science Foundation

Abstract

Changeover costs (and times) are central to numerous manufacturing operations. These costs arise whenever work centers capable of processing only one product at a time switch from the manufacture of one product to another. Although many researchers have contributed to the solution of scheduling problems that include changeover costs, due to the problem's combinatorial explosiveness, optimization-based methods have met with limited success. In this paper, we develop and apply polyhedral methods from integer programming for a dynamic version of the problem. Computational tests with problems containing one to five products (and up to 225 integer variables) show that polyhedral methods based upon a set of facet inequalities developed in this paper can effectively reduce the gap between the value of an integer program formulation of the problem and its linear programming relaxation (by a factor of 94 to 100 per cent). These results suggest the use of a combined cutting plane/branch and bound procedure as a solution approach. In a test with a five product problem, this procedure, when compared with a standard linear programming-based branch and bound approach, reduced computation time by a factor of seven.

Production planning and scheduling systems have a significant impact on the performance of manufacturing operations and, hence, have been the subject of extensive research. A question that has received considerable attention is how to schedule a production facility that processes one product at a time and incurs a changeover cost (and/or time) whenever it switches from the manufacture of one product to another. This problem has become a prototypical model in the operations management literature because it represents planning and scheduling decisions in numerous contexts. For example, in a paper mill, a paper machine incurs changeover costs (and time) when it switches from processing one grade of paper to another. These costs arise because a machine produces inferior quality paper and incurs breakage until it can be fully adjusted for the new grade. In the paint industry, the blending machines incur changeover costs between a change of colors due to the loss in production time and the use of expensive materials to clean the equipment. This model also describes certain assembly line operations, when they manufacture a few models of the same product on common assembly lines and incur substantial cost for setting up a line for a particular model. The Kool King case study in Marshall et al. (1975) describes the production of different models of air conditioners on such a facility.

Even though this scheduling problem has attracted much attention, researchers and practitioners can solve it for only special cases. Existing research has focused on the constant demand, infinite horizon version of the problem and the heuristics developed for this special case do not perform well for the general dynamic, deterministic problem.

In this paper, we study the polyhedral structure of an integer programming formulation of the dynamic, deterministic problem. The objective is to use these results to obtain improved formulations of the problem as well as to develop efficient solution methods for the general problem. This research is in part motivated by increasing empirical evidence that suggests that both pure and mixed integer programming problems can be solved to optimality in reasonable computation times by methods that use results about the underlying polyhedral structure of these problems. See, for example, the work of Crowder and Padberg (1980), Padberg and Hong (1980), Crowder, Johnson and Padberg (1983) and Johnson, Kostreva and Suhl(1985). Several of these studies address problems that arise in real world industrial applications. Recently, researchers have also investigated the polyhedral structure of the lot size model and reported considerable success using these results to solve larger problems that contain these models as substructures. See Barany, Van Roy and Wolsey (1984a, 1984b), Eppen and Martin (1985), Pochet (1986) and Leung, Magnanti and Vachani (1987). For additional references concerning the computational use of polyhedral methods, see the bibliography compiled by Grötschel (1985) and the survey by Hoffman and Padberg (1985).

This paper is organized as follows. The next section reviews the existing research on the changeover cost problem. The subsequent sections present an integer programming formulation for the problem, derive a class of non-trivial facets for the single product version of the problem and identify conditions under which these inequalities are also facets of the multiple product problem; they also show that our formulation of the problem is equivalent to a fixed charge network flow problem and that the facet inequalities can be interpreted as 'cut-set' inequalities for this network. The network flow interpretation provides further insight into the problem and also suggests additional valid inequalities. The final section describes a strong cutting plane algorithm for the problem that uses the facet inequalities to generate the cuts, and presents computational results. The computational results indicate that the facet inequalities significantly reduce the gap between the objective values of the original problem and its linear programming relaxation and lead to efficient solution techniques.

1. Literature

Much of the research to date, which has assumed constant demand and an infinite planning horizon, has attempted to develop a cyclic schedule that minimizes average costs. Elmaghraby (1978) calls this special case of the problem the Economic Lot Scheduling Problem (ELSP) and surveys methods that researchers have proposed for its solution. See, for example, Bomberger (1966), Stankard and Gupta (1969), Hodgson (1970), Doll and Whybark (1973) and Haessler (1979) for a description of some of the heuristics proposed to solve the ELSP. Although these heuristics provide reasonably good solutions to the ELSP, they do not perform well for the general case.

Several researchers have proposed dynamic programming methods for the general, deterministic version of the problem. Glassey (1968) and Tenzer (1969) considered the special case of unit changeover costs. Mitsumori (1972) and Gascon and Leachman (1985) extended Glassey's method to solve the problem with non-sequence dependent changeover costs. Driscoll and Emmons (1977) present an algorithm that allows sequence dependent changeover costs. The running time of all these algorithms increases exponentially with the number of products and the number of time periods and, hence, their use is limited to small problems.

Other researchers have proposed alternative approaches to solve the dynamic, deterministic problem. Geoffrion and Graves (1976) proposed a quadratic assignment approach to solve the multi-machine scheduling problem with sequence dependent changeover costs. Karmarkar, Kekre and Kekre (1987) discuss solution methods for the single item version of the problem. Schrage (1982) suggests a linear programming based method that is similar to the approach proposed by Manne (1958) for the lot size problem. Recently, Karmarkar and Schrage (1985) have proposed a formulation for the single facility problem that is similar to the one we study in this paper, and have discussed its solution using Lagrangean relaxation. However, they report that their computational results are not very encouraging. Eppen and Martin (1985) have reformulated the single item uncapacitated problem with both setup and changeover costs as a shortest path problem and obtained integer solutions to the original problem by solving the shortest path reformulation. They report good computational results for the multiple item capacitated problem using the the single item reformulation as a substructure.

Vergin and Lee (1978), Graves (1980) and Leachman and Gascon (1985a, 1985b) have studied the stochastic version of the problem.

2. Problem Statement and Formulation

The scheduling problem studied in this paper is defined by several modeling assumptions.

A1: The changeover time is zero, i.e., the capacity available for production in any period is not affected by whether or not the schedule commences production for any product in that period. This assumption is reasonable when the changeovers are made outside the regular production hours, for example at the end of the regular shift, or when the changeover times are small.

A2: The setup of the machine can be maintained even if the machine is idle in a particular period. This assumption is reasonable in situations such as paint manufacturing since once the equipment has been cleaned for the production of a particular color, it can be used to blend that color even after an idle period.

A3: The changeover costs are not sequence dependent, i.e., the changeover cost in any period depends only on the product for which the facility is to be setup and not on the product it was manufacturing earlier.

The formulation we propose is robust in the following sense: modifications of it that incorporate changes to some of these assumptions are amenable to analysis similar to that obtained in this paper. Consequently, it is possible to extend results for the original model to obtain facets for the new models (see Vachani, 1986, for details).

A4: The facility is scheduled using the following "discrete production policy": in any period, the production level is either zero or equal to capacity in that period.

Schrage (1982) refers to this policy as an "all or nothing policy." An alternative continuous policy would permit the production in a period to vary between zero and the available capacity in that period. Though the discrete policy is obviously a restricted version of the continuous one, it is a reasonable production policy in several contexts. For example, in some situations it may be very expensive to run a production line at less than full capacity. Also, if demand is high and the facility is capacity constrained, the discrete policy may be a reasonable production plan. The discrete policy may also be easier to implement and control. Moreover, the cost of a discrete policy can be made as close as desired to the cost of a continuous policy by redefining the length of the time periods. Therefore, it is reasonable to assume that the facility uses a discrete production policy and we discuss only this case.

Problem Formulation

Let T denote the finite horizon over which the facility is to be scheduled and let P denote the number of products to be processed on the facility. We assume that the relevant costs for any production policy are the inventory holding costs, the changeover costs and the fixed costs for producing in any period. Let h'_{pt} be the inventory holding cost per unit of product p in period t, K_{pt} be the changeover cost that is incurred to setup the facility whenever production of product p commences in period t, and s_{pt} be the fixed cost (perhaps zero) to maintain the setup for product p in period t. Finally, let d'_{pt} be the demand for product p in period t and let c_p be the capacity of the production line for product p (since this model is not used for long range planning, it is reasonable to assume that the capacity is constant over the horizon).

The demand constraints for the problem can be formulated as follows.

$$\sum_{j=1}^{t} w_{pj} \geq (1/c_p) \sum_{j=1}^{t} d'_{pj} \qquad \textit{for all } p,t. \qquad (1)$$

In this expression, the decision variable $w_{pt} \in \{0, 1\}$ equals 1 if product p is produced in period t (at production level c_p) and 0 if it is not produced in the period. These constraints require that cumulative production for each product p up to any time period t must meet the cumulative demand for that product up to period t. Since $w_{pt} \in \{0,1\}$ for all p and t, the lefthand side of inequality (1) is integral and hence, it is possible to replace the righthand side by $\lceil (1/c_p) \Sigma d'_{pj} \rceil$, where $\lceil x \rceil$ denotes the smallest integer greater than or equal to x. Redefining the demand in each period as

$$d_{pt} = \lceil (1/c_p) \sum_{j=1}^{t} d'_{pj} \rceil - \lceil (1/c_p) \sum_{j=1}^{t-1} d'_{pj} \rceil$$

permits us to reformulate the demand constraints (1) as

$$\sum_{j=1}^{t} w_{pj} \geq \sum_{j=1}^{t} d_{pj} \qquad \textit{for all } p,t.$$

Further, since the (scaled) production w_{pt} in any period t cannot exceed 1, the resulting demands d_{pt} can be adjusted so that $d_{pt} \in \{0, 1\}$ for all p and t. For example, if $d_{pj} > 1$, then setting $d_{pj} = 1, d_{p,j-1} = d_{p,j-1} + (d_{pj} - 1)$ results in an equivalent formulation. If the new $d_{p,j-1}$ is now greater than 1, repeat the procedure until $d_{pt} \in \{0,1\}$ for all p and t or the problem becomes infeasible (when $d_{p1} > 1$).

These observations permit us to formulate the scheduling problem, which we call CSP (for Changeover Cost Scheduling Problem), as follows. The decision variables in this model are w_{pt}, which equals 1 if product p is produced in period t and 0 if it is not; y_{pt}, which equals 1 if the machine is setup for product p in period t and 0 if it is not; and z_{pt}, which is 1 if the machine is changed over to product p in period t and 0 otherwise.

CSP

$$\text{Minimize} \quad \sum_{t=1}^{T}\sum_{p=1}^{P}\{K_{pt}z_{pt} + s_{pt}y_{pt} + h_{pt}w_{pt}\} \tag{2}$$

$$\text{subject to} \quad \sum_{j=1}^{t} w_{pj} \geq \sum_{j=1}^{t} d_{pj} \qquad \text{for all } p,t \tag{3}$$

$$\sum_{j=1}^{T} w_{pj} = \sum_{j=1}^{T} d_{pj} \qquad \text{for all } p \tag{4}$$

$$w_{pt} - y_{pt} \leq 0 \qquad \text{for all } p,t \tag{5}$$

$$z_{pt} + y_{p,t-1} - y_{pt} \geq 0 \qquad \text{for all } p,t \tag{6}$$

$$\sum_{p=1}^{P} y_{pt} \leq 1 \qquad \text{for all } t \tag{7}$$

$$0 \leq w_{pt} \leq 1,\ 0 \leq y_{pt} \leq 1,\ 0 \leq z_{pt} \leq 1 \qquad \text{for all } p,t \tag{8}$$

$$w_{pt} \text{ integer},\ y_{pt} \text{ integer},\ z_{pt} \text{ integer} \qquad \text{for all } p,t. \tag{9}$$

Let $w = (w_{pt})$, $y = (y_{pt})$, and $z = (z_{pt})$ denote vectors of the corresponding decision variables. Constraints (4) specify that the ending inventory for each of the products be zero (any other closing inventory policy could be imposed instead). Forcing constraints (5) state that $y_{pt} = 1$ if product p is produced in period t. Constraints (6) require the changeover variable z_{pt} to equal 1 if the machine is setup for product p in period t but not in period t-1 (i.e., if the machine changes over to product p in period t from some other product or from not being setup for production). Constraints (7) restrict the machine to be setup for at most one product in any period. The demands $d_{pt} \in \{0, 1\}$ for all p and t, the objective function coefficients h_{pt} of w_{pt} are derived from the original inventory holding costs h'_{pt} and the constant term involving d'_{pt} in the objective function has been dropped.

Let v(CSP) denote the optimal value of the objective function for the problem. Similarly, for any problem Q, let v(Q) denote its optimal objective value. Orlin (1986) has shown that problem CSP is NP-complete. Thus, the problem is not likely to be solved by any polynomial time algorithm. CSP can, in fact, be solved as a shortest path problem in $O(\{(T+P)/P\}^P TP^3)$ (see Vachani, 1986). However, this procedure is not efficient if both T and P are reasonably large.

Let CSPL denote the linear programming relaxation of CSP, i.e., CSPL is obtained from CSP by omitting constraints (9). It is relatively straightforward to show that, in the worst case, the gap between the optimal values of CSP and CSPL can be arbitrarily large. Therefore, CSPL, the linear programming relaxation, may not provide a tight lower bound for the optimal value of CSP. If we use Lagrangean relaxation to solve CSP by relaxing either constraints (5) or (6), then the resulting subproblem can be transformed into a network flow problem which satisfies the integrality property (Geoffrion, 1974). Hence, the lower bound derived from relaxing either of these constraints will not be better than v(CSPL). Karmarkar and Schrage (1985) solved the continuous production policy version of this scheduling problem (a problem that differs from CSP in the formulation of the demand constraints (3) and (4) since the continuous policy allows production of product p in period t to be any amount between 0 and c_p, instead of either 0 or c_p) by relaxing constraints (7) and reported that the computational results were not very encouraging. Thus, in general, Lagrangean relaxation may not be a good solution strategy for problem CSP. We would, therefore, like to construct better solution strategies for the problem; the success of cutting plane procedures using facet inequalities in other problem domains prompts our study of the polyhedral structure of CSP.

3. Polyhedral Structure of the Single Product Model

Let SCSP denote the single product version of the scheduling problem ($P=1$). SCSP differs from CSP only by the exclusion of constraints (7) which are now included as a part of constraints (8). Our goal in characterizing the structure of SCSP is not to develop a cutting plane procedure for SCSP itself (since it can be solved as a shortest path problem in $O(T^2)$ time), but to develop insights into the multiple product problem. In fact, as shown later, the results for SCSP can be directly extended to CSP. We discuss the single product problem first because the arguments are simpler.

Let F(SCSP) denote the set of feasible solutions for problem SCSP, and let $\mathbb{C}$ denote the convex hull of F(SCSP). In any given instance of SCSP, let $t_1, t_2, ..., t_n$, denote the periods with nonzero demand, i.e., the total demand over the planning horizon is n units. We can assume without loss of generality that $T = t_n$, since there will be no production in periods beyond t_n in any feasible solution. To exclude uninteresting cases, we also assume that $t_1 \geq 2$, since, if for some $1 \leq j \leq n$, $t_1 = 1, t_2 = 2, ..., t_j = j$ and $t_{j+1} > j+1$, then $w_t = y_t = 1$, $1 \leq t \leq j$, for all feasible solutions, and $z_1 = 1 - y_0$, $z_t = 0$, $2 \leq t \leq j$ in any optimal solution (assuming $K_t > 0$). Therefore, the scheduling problem needs to be solved only for periods $j+1$ through T, with the first period of nonzero demand being period $j+2$ or later, which is equivalent to a problem with $t_1 \geq 2$.

Non-Trivial Facets

Consider the inequality

$$\sum_{t=1}^{j-1} w_t + \sum_{t \in Q} w_t + \sum_{t \in S} y_t + \sum_{t \in C} z_t \geq q \,, \qquad (PI)$$

with $t_{q-1}+1 \leq j \leq t_q$, $P = \{j, j+1, \ldots, t_q\}$, $Q \subseteq P$, $S \subseteq P$, $Q \cap S = \emptyset$ *and* $C = P \setminus \{Q \cup S\}$.

We refer to inequalities PI as *partitioning inequalities* since they partition the interval P into the sets Q, S and C. The inequality includes the w_t variables corresponding to the Quantity produced in the periods Q, the Setup variables in the periods S, and the Changeover variables in the periods C. Note that since $j \geq t_{q-1}+1$, the total production up to period j-1 must be at least q-1 to meet demand up to period t_{q-1}. Therefore, inequality PI states that if the total production up to period j-1 is exactly q-1, i.e., it cannot meet demand up to period t_q, then the facility must either produce in one of the periods in the set Q, or be set up in one of the periods in the set S, or incur a changeover in one of the periods in the set C. The next proposition shows that PI is valid for $\mathbb{C}$ if the index sets Q, S and C satisfy certain additional conditions.

Proposition 1. PI is a valid inequality for $\mathbb{C}$ if it satisfies the following conditions.

(i) $j \notin C$, i.e., z_j is not in the inequality, and

(ii) if $t \in Q$ then $t+1 \notin C$, i.e., if w_t is in the inequality, then z_{t+1} is not.

If $j > 1$, then the conditions of the proposition are also necessary for the inequality to be valid.

Proof. Condition (i) is necessary (if $j > 1$) because even if production up to period j-1 is exactly q-1, the facility can produce in period j with $z_j = 0$, i.e., without incurring a changeover in period j. For example, the inequality

$$\sum_{t=1}^{j-1} w_t + z_j \geq q$$

with $j = t_q$ is not valid since a feasible solution with $y_t = 1$ for all $1 \leq t \leq t_q$, $w_t = 1$ for $t = t_1, t_2, ..., t_q$, and $z_1 = 1\text{-}y_0$, $z_t = 0$ for $2 \leq t \leq t_q$, violates it. Similarly, any version of this inequality with $t \in Q$ and $t+1 \in C$, e.g., $w_1 + z_2 \geq 1$ for $j = 1$ and $t_1 = 2$, is not valid since a feasible solution with $y_1 = y_2 = 1$, $w_1 = 0$, $w_2 = 1$, $z_1 = 1$ and $z_2 = 0$ violates it. If $y_0 = 0$ and $j = 1$, then condition (i) is not necessary since $z_1 + z_2 + ... + z_{t_1} \geq 1$ is a valid inequality. However, it is implied by $y_1 + z_2 + ... + z_{t_1} \geq 1$ since $z_1 \geq y_1$ in this case. Thus, even though the inequality may be valid it cannot be a facet unless condition (i) is satisfied since it is dominated by another valid inequality.

To establish that PI is valid if it satisfies both the conditions, let (w, y, z) be any feasible solution for SCSP. Since $\Sigma_1^{j-1} w_i \geq q - 1$ to meet demand, PI is trivially satisfied unless $\Sigma_1^{j-1} w_i = q\text{-}1$. In this case, the inequality states

$$\sum_{t \in Q} w_t + \sum_{t \in S} y_t + \sum_{t \in C} z_t \geq 1 .$$

Since $\Sigma_1^{t_q} d_i = q$ and $\Sigma_1^{j-1} w_i = q\text{-}1$, the facility must produce in at least one period between j and t_q. Let i be the first period between j and t_q for which $y_i = 1$ and let $a \geq i$ be the first period for which $w_t = y_t = 1$. If $i \notin S \cup C$ or $a \in Q \cup S$, then PI is satisfied (if $i \in C$, then by condition (i), $i > j$ and z_i must be 1 since $y_{i-1} = 0$). So consider $i \in Q$ and $a \in C$, i.e., w_i and z_a are in the inequality. By condition (ii), for some $i+1 \leq b \leq a\text{-}1$, $y_b, z_{b+1}, z_{b+2}, ..., z_a$ are all in the inequality. If $y_b = 1$, then PI is satisfied. If $y_b = 0$, then since $y_a = 1$, at least one of $z_{b+1}, z_{b+2}, ..., z_a$ equals 1 and PI is again satisfied. ∎

Theorem 1. The partitioning inequality PI is a facet of **C** if and only if Q, S and C satisfy the following conditions.

(i) $j \notin C$, i.e., z_j is not in the inequality,

(ii) if $t \in Q$ then $t+1 \notin C$, i.e., if w_t is in the inequality, then z_{t+1} is not,

(iii) if $t \in S$ then $t+1 \in C$, i.e., if y_t is in the inequality then so is z_{t+1} and, in particular, $t_q \notin S$,

(iv) if $Q = P$, then $q \neq n$ and $t_{q+1} > t_q + 1$, i.e., $d_{t_q+1} = 0$,

(v) if $q = n$, then $|S| = 1$.

Proof.

Necessity of the Conditions

Proposition 1 shows that (i) and (ii) are necessary for the inequality to be valid (or a facet if $j = 1$). To establish necessity of the other conditions, we show that if a particular condition is not satisfied, then PI is dominated by some other valid inequality I, i.e., I implies PI, but the converse is not true.

If y_t is in the inequality but z_{t+1} is not, then, by Proposition 1, y_t can be replaced by w_t to obtain another valid inequality I. Since $w_t \leq y_t$ for all t (and $w_t \neq y_t$ for all feasible solutions), I dominates PI and hence, condition (iii) is necessary for PI to be a facet.

If $Q = P$, then PI is the same as the demand constraint (3). If $q = n$, then (4) requires that PI be satisfied at equality and hence, PI cannot be a facet. If $d_{t_q+1} = 1$, then the demand constraint for period t_q+1 implies the demand constraint for period t_q, i.e., implies PI, since $w_t \leq 1$ for all t. Therefore, condition (iv) is necessary as well.

If $q = n$, then condition (iv) requires that $Q \neq P$ and hence, $|S| \geq 1$ (since $j \notin C$). Suppose $|S| > 1$. Consider the following inequality with $|S| = 2$ and $t_q = j+3 = T$.

$$I_{j-1} + y_j + z_{j+1} + y_{j+2} + z_{j+3} \geq 1. \quad (10)$$

Let $\mathbf{C^*} = \{(w, y, z) \in \mathbf{C} \mid (w, y, z)$ satisfies PI at equality$\}$ and let (w, y, z) be any point in $\mathbf{C^*}$. Since $j+3 = T$, $I_{j-1} + w_j + w_{j+1} + w_{j+2} + w_{j+3} = 1$ for all feasible solutions to SCSP, and hence, $w_{j+2} + w_{j+3} \leq 1$. If $w_{j+2} + w_{j+3} = 0$, then $I_{j-1} + w_j + w_{j+1} = 1$ to satisfy demand in period $j+3$, and consequently, $I_{j-1} + y_j + z_{j+1} = 1$ and $y_{j+2} + z_{j+3} = 0$. Similarly, if $w_{j+2} + w_{j+3} = 1$, then $y_{j+2} + z_{j+3} = 1$. Therefore, for any point (w, y, z) in $\mathbf{C^*}$, $w_{j+2} + w_{j+3} = y_{j+2} + z_{j+3}$. However, since all feasible solutions to SCSP that satisfy $w_{j+2} + w_{j+3} = y_{j+2} + z_{j+3}$ are not necessarily in $\mathbf{C^*}$, (10) cannot be a facet. This argument can be generalized to any other inequality with $|S| > 1$. Therefore, if $q = n$, then $|S| = 1$ for PI to be a facet.

Sufficiency of the Conditions

For any j, $1 \leq j \leq T$, it is easy to construct a feasible solution with $I_{j-1} = 0$: the machine is off in periods $j, j+1, \ldots, t_q - 1$, i.e., $w_t = y_t = z_t = 0$ for all $j \leq t \leq t_q - 1$, and the machine produces one unit in period t_q, i.e., $w_{t_q} = y_{t_q} = z_{t_q} = 1$. Since this solution satisfies PI at equality, PI is a face of $\mathbb{C}$. Similarly, it is straightforward to construct another feasible solution with the machine on in all periods and the changeover variable also equal to 1 in all periods, i.e., $y_t = z_t = 1$ for all and the machine produces in periods $1, t_1, t_2, \ldots, t_{n-1}$. This solution has $I_{j-1} = 1$ for all $j \geq 2$ (and $w_1 = y_1 = 1$) and $w_{t_k} = y_{t_k} = z_{t_k} = 1$ for all $k \leq n - 1$. Thus, $(w, y, z) \notin \mathbf{C^*}$ unless $q = n$ and $Q = P$. However, by condition (iv), if $Q = P$, then $q \neq n$. Therefore, $(w, y, z) \notin \mathbf{C^*}$ and $\dim \mathbf{C^*} \leq \dim \mathbb{C} - 1$. To prove that $\dim \mathbf{C^*} = \dim \mathbb{C} - 1$, let $\alpha w + \beta y + \gamma z = \delta$ represent an arbitrary equation that is satisfied by all $(w, y, z) \in \mathbf{C^*}$, where $\alpha \in \mathbb{R}^T$, $\beta \in \mathbb{R}^T$, $\gamma \in \mathbb{R}^T$ and $\delta \in \mathbb{R}$. We show that $\alpha w + \beta y + \gamma z = \delta$ must be a linear combination of

$$\sum_{t=1}^{j-1} w_t + \sum_{t \in Q} w_t + \sum_{t \in S} y_t + \sum_{t \in C} z_t = q \quad (11)$$

$$and \sum_{t=1}^{T} w_t = \sum_{t=1}^{T} d_t \qquad (12)$$

which are satisfied by all points in $\mathbb{C}^*$. The proof constructs a sequence of points (w^1, y^1, z^1), (w^2, y^2, z^2), ..., all in $\mathbb{C}^*$, and uses the fact that $\alpha w^1 + \beta y^1 + \gamma z^1 = \alpha w^2 + \beta y^2 + \gamma z^2 = \ldots$ to prove that (i) $\gamma_t = 0$ for all $t \notin C$, (ii) $\beta_t = 0$ for all $t \notin S$, (iii) $\alpha_t = \underline{\alpha}$ for all $1 \le t \le j\text{-}1$ or $t \in Q$, $\alpha_t = \alpha^*$ for all other t, and (iv) $\delta = \underline{\alpha}q + \alpha^* (\Sigma_1^T d_i - q)$. These conclusions establish the desired result. To prove that $\gamma_t = 0$ for a given t requires the construction of two solutions in $\mathbb{C}^*$, one with $z_t = 0$, the other with $z_t = 1$, and all other variables the same for both solutions. Note that if $y_t - y_{t-1} \le 0$, then forcing constraint (6) allows z_t to take value 0 or 1. We make use of this fact to construct the two solutions. Similarly, if it is possible to construct a pair of solutions in $\mathbb{C}^*$, one with $y_t = 0$, the other with $y_t = 1$, and all other variables the same for both solutions, then β_t must be 0. Defining $w_t = 0$ and $z_t = 1$ allows y_t to be either 0 or 1. To prove that $\alpha_i = \alpha_t$ for $i \neq t$ requires two solutions in $\mathbb{C}^*$, one with $w_i = 0$, $w_t = 1$, the other with $w_i = 1$, $w_t = 0$, and all other variables the same in both solutions. We now provide the details of the proof.

(1) $\gamma_t = 0$ *for all* $t \notin C$ *and* $\beta_t = 0$ *for all* $t \notin S$.

Since $t_1 \ge 2$, it is easy to construct a solution in $\mathbb{C}^*$ with $y_t = z_t = 0$ for any $t \le j\text{-}1$. Setting $z_t = 1$ in this solution and keeping all other variables the same yields another solution in $\mathbb{C}^*$ and hence $\gamma_t = 0$. Similarly, setting both $y_t = 1$ and $z_t = 1$ instead of 0 in the original solution still keeps the solution in $\mathbb{C}^*$ and shows that $\beta_t = 0$ as well (since $\gamma_t = 0$). Thus, $\gamma_t = \beta_t = 0$ for all $t \le j - 1$. Now consider $t \ge t_q + 1$. If $d_{t_q+1} = 0$, then as for $t \le j\text{-}1$, it is easy to construct a solution in $\mathbb{C}^*$ with $y_t = 0$ for any $t \ge t_q+1$. If $d_{t_q+1} = 1$, then condition (iv) of the Theorem requires that $Q \neq P$, i.e., $S \neq \emptyset$. Let $b \in S$. In this case, construct a solution in $\mathbb{C}^*$ with $I_{j-1} = 0$, $w_b = w_{b+1} = 1$, $y_b = y_{b+1} = 1$ (i.e., $I_{t_q} = 1$), $z_{b+1} = 0$ and $y_t = 0$ for any $t \ge t_q+1$. Therefore, $\gamma_t = \beta_t = 0$ for all $t \le j - 1$ or $t \ge t_q + 1$.

Constructing another solution in $\mathbb{C}^*$ with $I_{j-1} = 1$ and $w_t = y_t = z_t = 0$ for all $t \in P$ shows that $\gamma_t = \beta_t = 0$ for all $t \in Q$ and $\gamma_t = 0$ for all $t \in S$. (If $j = 1$, the argument can easily be modified.) To show that $\beta_t = 0$ for $t \in C$, let $b \in C$. By definition of PI, either $b\text{-}1 \in S$ or $b\text{-}1 \in C$. Suppose $b\text{-}1 \in S$. We can now construct a solution in $\mathbb{C}^*$ with $I_{j-1} = 0$, $w_{b-1} = y_{b-1} = z_{b-1} = 1$ and $w_t = y_t = z_t = 0$ for all other $j \le t \le t_q$. Changing y_b to 1 instead of 0 in this solution keeps the solution in $\mathbb{C}^*$ (since $b \in C$) and shows that $\beta_b = 0$. A similar construction shows that $\beta_t = 0$ if $t\text{-}1 \in C$. Therefore, $\beta_t = 0$ for all $t \in C$.

(2) $a_t = \underline{a}$ *whenever* $1 \le t \le j\text{-}1$ *or* $t \in Q$, *and* $a_t = a^*$ *for all other* t.

Since $t_1 \ge 2$, construct a solution in $\mathbb{C}^*$ with $w_1 = 0$ and $w_t = 1$ for a given t, $2 \le t \le j\text{-}1$ or $t \in Q$. Letting $w_1 = 1$ and $w_t = 0$ in this solution and keeping all other variables the same yields another solution in $\mathbb{C}^*$ and shows that $\alpha_t = \alpha_1$ for $1 \le t \le j - 1$ or $t \in Q$. If $Q = P$, then $d_{t_q+1} = 0$ and a similar argument shows that $\alpha_t = \alpha^*$ for all other t. Now suppose that $Q \neq P$ and $q \neq n$. As in the proof of (1), for any $t \in S \cup C$ and $t+1 \in C$, construct a solution in $\mathbb{C}^*$ with $y_t = y_{t+1} = 1$. The form of this solution allows the shifting of a unit of production from any period after t_q to either of the periods t or t+1. Hence, $\alpha_t = \alpha^*$ for all $t \in S \cup C$ or $t \ge t_q+1$. If $Q \neq P$ but $q = n$, then $|S| = 1$ and we can similarly show that $\alpha_t = \alpha^*$ for all $t \in S \cup C$.

Therefore, $\alpha w + \beta y + \gamma z = \delta$ is of the form

$$\underline{\alpha}\sum_{t=1}^{j-1} w_t + \underline{\alpha}\sum_{t\in Q} w_t + \sum_{t\in S}\beta_t y_t + \sum_{t\in C}\gamma_t z_t + \alpha^* \sum_{t\notin Q,\, t\ge j} w_t = \delta. \tag{13}$$

Constructing a solution in $\mathbb{C}^*$ with $\Sigma_1^{j-1} w_i = q$ and $w_t = y_t = z_t = 0$ for all $t \in P$ shows that $\delta = \underline{\alpha}q + \alpha^*(\Sigma_1^T d_i - q)$. Constructing another solution in $\mathbb{C}^*$ with $\Sigma_1^{j-1} w_i = q - 1$ and $w_t = y_t$

$= z_t = 1$ for some $t \in P, t \notin Q$ shows that $\beta_t = \underline{\alpha} - \alpha^*$ for all $t \in S$ and $\gamma_t = \underline{\alpha} - \alpha^*$ for all $t \in C$. Therefore, (13) is equivalent to

$$\alpha'\{\sum_{t=1}^{j-1} w_t + \sum_{t \in Q} w_t + \sum_{t \in S} y_t + \sum_{t \in C} z_t\} + \alpha^* \sum_{t=1}^{T} w_t = \alpha' q + \alpha^* \sum_{t=1}^{T} d_t \qquad (14)$$

where $\alpha' = \underline{\alpha} - \alpha^*$. (14) is a linear combination of (11) and (12). Hence, $\dim \mathbb{C}^* = \dim \mathbb{C} - 1$, and PI is a facet of $\mathbb{C}$. ∎

Trivial Facets

Arguments similar to those used to prove Theorem 1 establish the following results.

Proposition 2. The constraint $w_j - y_j \leq 0$ is a facet of $\mathbb{C}$ for all $1 \leq j \leq T$.

Proposition 3. The inequality $z_j + y_{j-1} - y_j \geq 0$ is a facet of $\mathbb{C}$ for all $1 \leq j \leq T$ if the following two conditions are satisfied (i) if $j = 1$, then $y_0 = 0$, and (ii) $t_q \geq q + 2$, where $t_{q-1} + 1 \leq j \leq t_q$. (Note that $t_1 \geq 2$ implies $t_q \geq q + 1$ for all q.) If $t_q = q + 1$, then the inequality $z_j + y_{j-1} - y_j \geq 0$ can be replaced by the stronger inequality $z_j + y_{j-1} - 1 \geq 0$ which is a facet of $\mathbb{C}$ for this case.

Proposition 4. The demand constraint $\Sigma_1^j w_i \geq \Sigma^j d_i$ is a facet of $\mathbb{C}$ if $j \in \{t_1, t_2, ..., t_{n-1}\}$ and $d_{j+1} = 0$. If $d_{j+1} = 1$ or if $j \notin \{t_1, t_2, ..., t_{n-1}\}$, then the demand constraint is redundant. If $j = t_n = T$, then the constraint $\Sigma_1^T w_i \geq \Sigma^T d_i$ is an improper face of $\mathbb{C}$.

Proposition 5. The constraint $y_j \leq 1$ is a facet of $\mathbb{C}$ for all $2 \leq j \leq T$; $y_1 \leq 1$ is also a facet if $y_0 = 1$. (If $j = 1$ and $y_0 = 0$, then $z_1 \leq 1$ and $z_1 \geq y_1 - y_0 = y_1$ together imply $y_1 \leq 1$. Hence, $y_1 \leq 1$ cannot be a facet in this case.)

Proposition 6. The constraint $z_j \leq 1$ is a facet of $\mathbb{C}$ for all $1 \leq j \leq T$.

Proposition 7. For any j, $1 \le j \le T$, the constraint $w_j \ge 0$ is a facet of $\mathbb{C}$ if the demand period t_q, defined by $t_{q-1} + 1 \le j \le t_q$, satisfies $t_q \ge q + 2$. (If $t_q = q + 1$, then $w_j \ge 0$ cannot be a facet since it is implied by $\Sigma_1^{q+1} w_i \ge \Sigma_1^{q+1} d_i = q$ and $w_i \le 1$ for all i.)

Proposition 8. The constraint $z_j \ge 0$ is a facet of $\mathbb{C}$ for all $2 \le j \le T$; $z_1 \ge 0$ is also a facet if $y_0 = 1$. (If $j = 1$ and $y_0 = 0$, then $y_1 \ge 0$ and $z_1 \ge y_1 - y_0 = y_1$ together imply $z_1 \ge 0$. Hence, $z_1 \ge 0$ cannot be a facet in this case.)

Note that the remaining constraints, $y_j \ge 0$ and $w_j \le 1$, are redundant for SCSP and obviously cannot be facets of $\mathbb{C}$.

4. Interpreting the Single Product Model as a Fixed Charge Network Flow Problem

This section shows that the single product scheduling problem SCSP is equivalent to a fixed charge network flow problem and that the partitioning inequalities PI can be interpreted as "cut-set" inequalities for this network. Two observations motivate this line of inquiry. First, it may enable us to use results from the network design literature to gain further understanding of this scheduling problem as well as allow us to explore solution methods that have been used successfully to solve general network design problems (for example, see Balakrishnan, Magnanti and Wong (1987)). Second, our results concerning the structure of the scheduling problem may provide new insights about the general network design problem or about other special versions of the general problem.

To show that problem SCSP is equivalent to a fixed charge network flow problem, consider a 5 period example with unit demand in periods 2 and 5. It is easy to extend the arguments used to construct the network corresponding to this example to the general problem. Figure 1 shows the network for the 5 period problem. The dotted arcs in the network represent arcs with fixed charges and the solid arcs represent arcs with no fixed costs. The network has 3T+1 nodes (16 in this case), including a single source node S with supply of $\Sigma_1^T d_t$ units (2 in this example). Nodes t, t' and t" in the network correspond to period t in the original problem. Only nodes t can have positive demand; all other nodes, other than node S, are transshipment nodes (i.e., have 0 demand and supply). In this example, nodes 2 and 5 have positive demand, corresponding to the periods with positive demand in the original problem.

The flow on arc (t',t) is w_t and corresponds to production in period t in the original problem. The cost per unit flow on this arc is h_t and the arc has a capacity of 1 unit. The flow on arc (t,t+1) is I_t, the ending inventory in period t, with 0 cost per unit flow since h_t includes the inventory holding cost. Variable y_t is an indicator variable that equals 1 if the solution sends any positive flow on arc (t",t') and equals 0 otherwise. Thus, if y_t is 1, the solution incurs a fixed charge of s_t for flow on arc (t",t'). Note that if $w_t > 0$, i.e., if the solution sends flow on arc (t',t), then it must also send flow on arc (t",t') since all paths from S to t' include arc (t",t'). Thus, $y_t = 1$ if $w_t > 0$, which is equivalent to the forcing constraint (5) of the original problem. Similarly, z_t is an indicator variable that equals 1 if the solution sends any positive flow on arc (S,t") and, therefore, incurs a fixed cost of K_t. From the network construction, it is again easy to see that if $y_t = 1$, i.e., the solution sends flow on arc (t",t'), then it must send flow on either arc (S,t") or (t"-1,t'-1), i.e., either $z_t = 1$ or $y_{t-1} = 1$ or both, which is the "changeover variable" requirement (7) imposed in SCSP. The variable cost of unit flow is 0 on all arcs other than arcs (t',t) and the model imposes no upper bound on the flow in any of the arcs other than arcs (t',t).

These observations establish the equivalence between SCSP and NP, the network flow problem described in Figure 1. To interpret the partitioning inequalities for this network

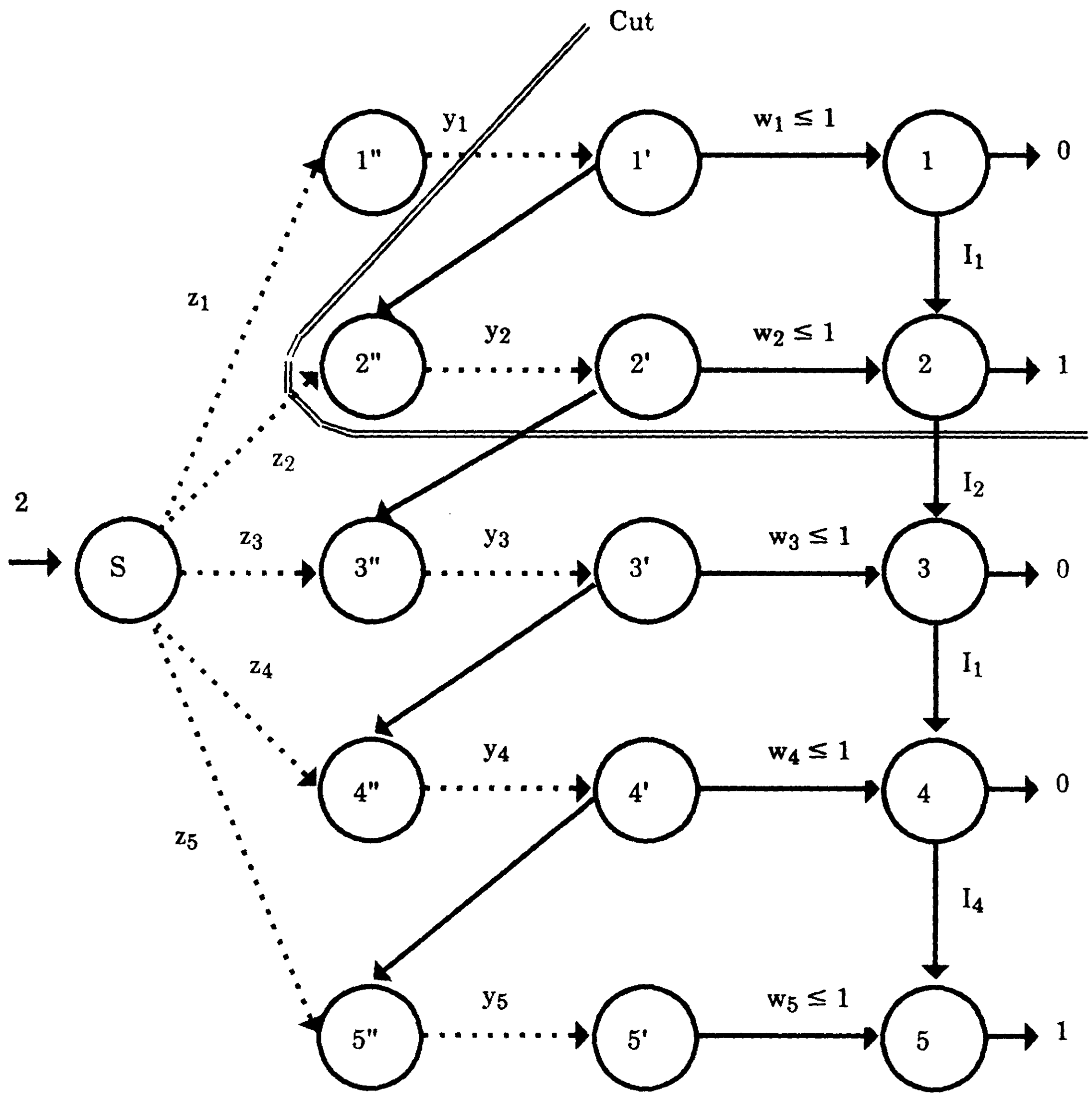

Figure 1 Network Corresponding to Single Product Scheduling Problem

flow problem, let f_{ij} represent the flow on arc (i,j) in any solution to NP and let N denote the set of nodes in the network design model. Figure 1 also shows a cut-set (X,X') with X' = {1, 2, 1', 2', 2"} and $X = N \setminus X'$. Since the set X' has 1 unit of demand that must be met by flow from X to X', the following inequality is valid for all solutions to the problem.

$$f_{1"1'} + f_{S2"} - f_{2'3"} - I_2 \geq 1$$

$$\Rightarrow \; f_{1"1'} + f_{S2"} \geq 1$$

$$\Rightarrow \; y_1 + z_2 \geq 1. \qquad (15)$$

Similarly, the cut-set (Y, Y') with Y' = {3', 3, 4", 4', 4, 5", 5', 5} and $Y = N \setminus Y'$ yields the following valid inequality for the problem.

$$I_2 + f_{3"3'} + f_{S4"} + f_{S5"} \geq 1$$

$$\Rightarrow \; I_2 + y_3 + z_4 + z_5 \geq 1$$

$$\Leftrightarrow \; w_1 + w_2 + y_3 + z_4 + z_5 \geq 2, \qquad (16)$$

since $I_2 = w_1 + w_2 - 1$. Both inequalities (15) and (16) correspond to partitoning inequalities for SCSP. Note that both these inequalities (as well as the general partitioning inequalities) are derived from a special type of cut (W, W') in the network -- node $t_q \in W'$ for some q and the other nodes in W' are a subset of the nodes that correspond to periods $j, j+1, ..., t_q$ with $t_{q-1}+1 \leq j \leq t_q$. This observation suggests that it may be possible to generalize the partitioning inequalities by examining other types of cut-sets for NP.

Generalizing the Partitioning Inequalities

Consider (Z, Z'), with Z = {S, 1"} and $Z' = N \setminus Z$, a different type of cut-set for the 5-period single product example. From this cut-set, we obtain the following valid inequality for NP

$$f_{1"1'} + f_{S2"} + f_{S3"} + f_{S4"} + f_{S5"} \geq 2 \qquad (17)$$

since the set Z' has a demand of 2 units that must be met by flow from Z to Z'. Note that the flow in either of the arcs (1",1') or (S,2") used to meet demand in the set Z' cannot exceed 2 units since Z' has a demand of only 2 units. Therefore, $f_{1"1'} \leq 2y_1$ and $f_{S2"} \leq 2z_2$. Similarly, $f_{S3"} \leq z_3$, $f_{S4"} \leq z_4$ and $f_{S5"} \leq z_5$ and we can replace (17) by

$$2y_1 + 2z_2 + z_3 + z_4 + z_5 \geq 2. \qquad (18)$$

Note, though, that this inequality is implied by the partitioning inequality $y_1 + z_2 \geq 1$. However, a different procedure for replacing the flow variables in (17) by the design variables y and z yields a new inequality. We first show that

$$f_{1''1'} + f_{S2''} \leq y_1 + z_2 + y_2. \qquad (19)$$

If $f_{1''1'} = 1$ and $f_{S2''} = 0$, then $f_{1''1'} + f_{S2''} \leq y_1$. Now suppose that $f_{1''1'} = 2$. Since arc (1',1) has a capacity of 1 unit, at least one unit of the flow in (1'',1') must flow through arc (2'',2'). Hence, if $f_{1''1'} = 2$, both y_1 and y_2 must equal 1 and $f_{1''1'} \leq y_1 + y_2$. Similarly, if $f_{S2''} > 0$, then both z_2 and y_2 must equal 1. However, since $f_{1''1'} + f_{S2''} \leq 2$ in all cases, (19) is valid. Therefore, (18) can be replaced by

$$y_1 + z_2 + y_2 + z_3 + z_4 + z_5 \geq 2. \qquad (20)$$

Inequality (20) is not implied by any of the partitioning inequalities; in fact, it is a facet for the convex hull of solutions to this particular problem instance. This discussion suggests that we can derive valid inequalities for the general problem SCSP that are similar to (20).

Consider the inequality

$$\sum_{t=1}^{j-1} w_t + \sum_{t \in Q1} w_t + \sum_{t \in S1} y_t + \sum_{t \in C1} z_t + \sum_{t \in Q2} w_t + \sum_{t \in S2} y_t + \sum_{t \in C2} z_t \geq q+1 \qquad (21)$$

$$\text{with } t_{q-1}+1 \leq j \leq t_q,\ P1 = \{j, j+1, \ldots, t_q\},\ P2 = \{t_q+1, t_q+2, \ldots, t_{q+1}\},$$

$$Q1 \cup S1 \cup C1 = P1,\ Q1 \cap S1 = \varnothing, Q1 \cap C1 = \varnothing,$$

$$Q2 \subseteq P2,\ S2 \subseteq P2,\ Q2 \cap S2 = \varnothing \text{ and } C2 = P2 \backslash \{Q2 \cup S2\}.$$

Note that for any j and $t_{q-1} + 1 \leq j \leq t_q$, this inequality considers two sets of periods -- $P1 = \{j, j+1, \ldots, t_q\}$ and $P2 = \{t_q+1, t_q+2, \ldots, t_{q+1}\}$. The set P1 is divided into the subsets Q1, S1 and C1 and the set P2 is divided into Q2, S2 and C2. Note, further, that we do not require $S1 \cap C1 = \varnothing$. Inequality (20) is a special case of (21) with $j = 1$, $Q1 = Q2 = \varnothing$, $S1 = \{1, 2\}$, C1

= {2} and C2 = {3, 4, 5}. Also note that if Q1 = P1, then inequality (21) is the same as the partitioning inequality, i.e., the partitioning inequalities are a special case of inequalities (21).

Arguments similar to those used to show that (20) is valid for the example of Figure 1, establish the following proposition. A number of the conditions of this proposition are similar to the conditions of Proposition 1.

Proposition 9. Inequality (21) is valid for SCSP if the following conditions are satisfied.

(i) $j \notin C1$, i.e., z_j is not in the inequality,

(ii) if $t_q + 1 \in C2$, then $t_q \in S1$ and C1, i.e., if z_{t_q+1} is in the inequality, then so are y_{t_q} and z_{t_q},

(iii) for $t \in P1$, if $t \in C1$ only, then $t\text{-}1 \in S1$ and C1, i.e., for $t_{q\text{-}1} + 1 \leq t \leq t_q$, if z_t is in the inequality (but y_t is not), then both $z_{t\text{-}1}$ and $y_{t\text{-}1}$ are in the inequality,

(iv) if $t \in Q1 \cup Q2$, then $t+1 \notin C1 \cup C2$, i.e., if w_t is in the inequality, then z_{t+1} is not.

Some additional conditions, similar to those of Theorem 1 for the partitioning inequalities, ensure that inequality (21) is also a facet of SCSP. However, we do not discuss these conditions since inequality (21) can be generalized further and we address this question in our future research . It is important to pursue this line of inquiry further for two reasons. First, it will identify additional facets for this problem and second, it will provide insight into how to strengthen cut-set inequalities, similar to (17), for the general network design problem.

5. Structure of the Multiple Product Model

This section shows that results for the single product model can be extended to the multiple product problem. As in our discussion for the single product model, let t_{p1}, t_{p2}, ...,

t_{pn_p} denote the periods with positive demand for product p. Further, let C_m denote the convex hull of solutions to PCP, i.e., C_m = conv {(w, y, z) | (w, y, z) satisfies (3) - (9)}. The partitioning inequalities PI are still valid for PCP and for any product b can be stated as

$$\sum_{t=1}^{j-1} w_{bt} + \sum_{t \in Q} w_{bt} + \sum_{t \in S} y_{bt} + \sum_{t \in C} z_{bt} \geq q \qquad (PI_m^b)$$

$$with \;\; t_{b,q-1}+1 \leq j \leq t_{bq},\;\; P = \{j, j+1, \ldots, t_{bq}\},\;\; Q \subseteq P,\;\; S \subseteq P,\;\; Q \cap S = \emptyset \; and \; C = P \backslash \{Q \cup S\}.$$

As in the single product model, to exclude uninteresting special cases, we assume that a feasible solution with $y_{p1} = 0$ for all p, i.e., the demand pattern of the products permits the machine to be off in period 1. (A necessary and sufficient condition for this solution to exist is $\Sigma_p \Sigma_{\tau \leq t} d_{p\tau} \leq (t\text{-}1)$ for all t.) Arguments similar to those used to prove Theorem 1 establish the next theorem.

Theorem 2. PI_m^b is a facet of C_m if and only if Q, S and C satisfy the following conditions.

(i) $j \notin C$, i.e., z_{bj} is not in the inequality,

(ii) if $t \in Q$, then $t+1 \notin C$, i.e., if w_{bt} is in the inequality, then $z_{b,t+1}$ is not,

(iii) if $t \in S$, then $t+1 \in C$, i.e., if y_{bt} is in the inequality, then so is $z_{b,t+1}$ and, in particular, $t_{bq} \notin S$,

(iv) if Q = P, then $q \neq n_b$ and $t_{b,q+1} > t_{bq} + 1$, i.e., $d_{b,t_q+1} = 0$,

(v) if $q = n_b$, then $|S| = 1$, and

(vi) if $t \in C$ and $t < t_{bq}$, then $\Sigma_p \Sigma_{\tau \leq t} d_{p\tau} \leq t\text{-}2$.

Proposition (9) concerning the more general inequality (21) and results regarding the trivial facets can be extended similarly; we omit the details.

6. Separation Problem

If we want to use the partitioning inequalities as part of a cutting plane procedure to solve CSP (or any problem that contains CSP as a subproblem), then given a fractional point (w, y, z) that is feasible for the linear programming relaxation of CSP (or the larger problem), we need to identify a partitioning inequality that cuts it off or determine that no such inequality exists. This problem is referred to as the separation problem and the following algorithm solves it.

Let (w, y, z) denote a given fractional solution that is feasible for the linear programming relaxation of CSP. For any given q and j, with $t_{q-1}+1 \le j \le t_q$ and $q \in \{1, 2, ..., n\}$, let $P = \{j, j+1, ..., t_q\}$. If $\Sigma_1^{j-1} w_t \ge q$, then (w, y, z) satisfies PI for all Q, S and C that are subsets of the given set P. Now suppose that $\Sigma_1^{j-1} w_t < q$. In that case, define Q*, S* and C* as follows:

$$\sum_{t \in Q^*} w_t + \sum_{t \in S^*} y_t + \sum_{t \in C^*} z_t = min_{Q,S,C} \{ \sum_{t \in Q} w_t + \sum_{t \in S} y_t + \sum_{t \in C} z_t \}$$

with the minimum of the righthand side determined over all sets Q, S and C that satisfy conditions (i) - (v) of Theorem 1 and let

$$f(t_q) = \sum_{t \in Q^*} w_t + \sum_{t \in S^*} y_t + \sum_{t \in C^*} z_t .$$

If $f(t_q) < q - \Sigma_1^{j-1} w_t$, then Q*, S* and C* identify a violated inequality. Otherwise, by definition of Q*, S* and C*, there is no violated inequality for the given q and j.

For a given j and q, the following procedure finds $f(t_q)$ (and Q*, S* and C*). For any i satisfying, $j \le i \le t_q$, define g(i, w) {g(i, y), g(i, z)} as

$$g(i,w) = \quad min_{W,Y,Z}\{\sum_{t\in W} w_t + \sum_{t\in Y} y_t + \sum_{t\in Z} z_t\}$$

where $W \cup Y \cup Z = \{j, j+1, \dots i\}$, $W \cap Y = \emptyset$, $W \cap Z = \emptyset$, $Y \cap Z = \emptyset$, $i \in W \{i \in Y, i \in Z\}$, and, in addition, W, Y and Z satisfy: (i) $j \notin Z$, (ii) if $t \in W$, then $t+1 \notin Z$, and (iii) if $t \in Y$, then $t+1 \in Z$. Since condition (i) requires that $j \notin Z$, define $g(j, z) = \infty$. By definition of $g(i,.)$,

$$g(j,w) = w_j, \quad g(j,y) = y_j, \quad g(j,z) = \infty,$$

and for $i>j$,

$$g(i,w) = \{w_i + min[g(i-1,w), g(i-1,z)]\},$$

$$g(i,y) = \{y_i + min[g(i-1,w), g(i-1,z)]\},$$

$$g(i,z) = \{z_i + min[g(i-1,y), g(i-1,z)]\},$$

$$and \;\; f(t_q) = min\{g(t_q,w),\; g(t_q,z)\}, \text{ since condition (iii) requires } t_q \notin Y.$$

If $f(t_q) < q - \Sigma_1^{j-1} w_t$, the sets Q*, S* and C* and the violated inequality can be constructed by backtracking. It is easy to see that given (w, y, z), and j and q, it takes O(T) time to compute $f(t_q)$ and construct a violated inequality if one exists for the given j and q. Since there are exactly T possible combinations of j and q, the procedure for checking if a given fractional point violates any partitioning inequality PI, and constructing a violated inequality if one exists, takes $O(T^2)$ time. A similar procedure solves the separation problem for the more general valid inequalities (21). Recall that the single product problem SCSP can be solved as a shortest path problem in $O(T^2)$ time. Therefore, we would not want to develop a strong cutting plane procedure to solve this version of the problem. The fact that we can solve the separation problem efficiently is important when we implement a cutting plane procedure for CSP or any larger problem that contains CSP as a substructure. The next section describes such an algorithm that uses the separation problem procedure to identify cuts to be added to the current linear programming relaxation.

7. Computational Results

Based on our characterization of the polyhedral structure of CSP, we implemented a strong cutting plane algorithm to solve the problem. The algorithm starts by solving CSPL, the linear programming relaxation of the given problem. If the linear programming solution is fractional, then the algorithm checks if this solution violates any of the facet inequalities of CSP. We use both the partitioning inequalities and a special class of the more general valid inequalities (21) to generate cuts to eliminate the fractional solution. The cuts are identified by solving the separation problem for these inequalities. If the algorithm finds such a violated inequality, it adds it to the current linear programming relaxation of the problem and then solves the updated linear program. Our implementation repeats this procedure until either the linear programming solution is integral or it violates none of the facet inequalities. On termination, if the linear programming solution is not integral, then the algorithm uses branch and bound to obtain an optimal integral solution to the original problem.

Our computational experiments have two major objectives: (i) to estimate empirically the reduction in the integrality gap, i.e., the gap between the optimal values of the original problem and its linear programming relaxation, both before and after the addition of facet inequalities, and (ii) to determine if any specific subclass of the facet inequalities is more effective in reducing this gap. Identifying such a subclass would provide insight into modeling these problems and suggest linear programming-based solution methods that include these inequalities a priori. Our goal is not to develop the most efficient cutting plane procedure that exploits our description of the problem's polyhedral structure; therefore, for each fractional solution encountered in the algorithm, we solve the separation problem, rather than use a faster heuristic, to identify a violated inequality. Moreover, instead of testing when it is best to add violated inequalities and when to invoke branch and bound, we use branch and bound only when no more valid inequalities can be added to tighten the linear program. A number of such implementation issues needs to be resolved in order to develop a computationally efficient algorithm.

We performed all the computations on a PRIME 850 computer using the LINDO mixed integer programming package for solving the linear programs and for conducting the branch and bound computations. The matrix generators and the cut generation routines were coded in FORTRAN as part of the USER subroutine available with LINDO.

Data Sets

The literature contains very little test data for this scheduling problem. Karmarkar and Schrage (1985) report computational experience for a similar problem, but with a continuous production policy instead of a discrete production policy (refer to the earlier sections for a discussion of the relationship between the two policies). They use Lagrangean relaxation to solve problem instances of up to 4 products and 8 time periods. Their method for generating the data allows any particular instance to be infeasible if the initial inventory is zero. Therefore, they modify the problem to allow for any amount of initial inventory to be available at a cost that will discourage the use of this inventory unless the problem is infeasible without it.

For model CSP, it is easy to check if a given instance of the problem, with any given initial inventory, is feasible or not. (If the initial inventory for all the products is zero, then the problem is feasible if and only if $\Sigma_p \Sigma_{i \le t} d_{pi} \le t$ for all t.) Therefore, for our computational study, we generate a set of feasible problem instances with zero initial inventory. Further, instead of allowing demand to vary arbitrarily and then scaling it in units of capacity to obtain $d_{pt} \in \{0, 1\}$ for all p and t, we set the capacity equal to 1 for all the problem instances and set demand equal to 0 or 1 in every period for each of the products. We generate a set of problems with 1, 2 or 5 products and 10, 15 or 20 time periods. The largest problem instance we tested was a 5-product, 15-period problem. The number of 0-1 variables in these problems is equal to 3PT, (P is the number of products and T is the length of the horizon). Therefore, for the problems we solved, the number of 0-1 variables varied between 30 and 225.

All the problem instances set the initial inventory for each of the products equal to 0 and assume the machine is off at the start of the horizon. The other parameters of the problem are determined as follows.

(i) The utilization of the facility varies between 30% and 100%. For example, a 10-period problem with 50% utilization has a total demand of 5 units for all the products.

(ii) The total demand for the facility is divided equally between the products (or as close to equal as possible since demand for the products is integral). Thus, a 2-product, 10-period problem with 50% utilization has a demand of 3 units for product 1 and 2 units for product 2. For each product, the periods with positive demand are distributed uniformly over the horizon, subject to the requirement that the resulting problem be feasible. We ensure feasibility of the problem instance by first generating the demand for product 1. We then determine the first period with positive demand for the next product. If the demands generated so far satisfy $\Sigma_p \Sigma_{i \leq t} d_{pi} \leq t$ for all t, then the problem is feasible and we generate the next period of positive demand for this product. If, however, $\Sigma_p \Sigma_{i \leq t} d_{pi} > t$ for some t, then it is not feasible to allow positive demand in the last period generated and we determine another period instead. Proceeding sequentially in this manner, we generate demand for all the products. If none of the products has positive demand in period T, the last period of the horizon, then we change the last period with positive demand for product 1 to equal T, thereby ensuring that at least one of the products has positive demand in the last period of the horizon.

(iii) The cost parameters are the same for all the products and are constant over the horizon. The inventory holding cost for all the problem instances is equal to 10/unit/period. Note that this cost of 10 implies that the coefficient h_t of w_t in problem CSP is 10(T-t), where T is the length of the horizon. The setup cost s_t is equal to 20 for all the single product problems and 0 for all the 2 and 5 product problems (we explain the reason for this difference later). For each combination of number of products, number of time periods and capacity utilization, we generate 3 problems with the changeover cost K_t equal to 50, 100 or 200. These choices permit us to determine whether the problem becomes harder to solve as the changeover cost increases.

Before presenting the results, we make a few observations about specific problem instances. Let $CSPL_1$ denote the linear programming relaxation of CSP with the following additional constraints

$$y_{p1} + z_{p2} + \ldots + z_{pt_{p1}} \geq 1 \quad \textit{for all } p$$

included. Recall that t_{p1} denotes the first period with positive demand for product p. For the single product version of the problem, *with* $s_t = 0$ *for all* t, it is easy to show that $v(CSPL_1) = v(CSP)$. Therefore, in this case, only a single additional constraint is needed to close the integrality gap to 0. To exclude this simple case from our computations, we let the setup cost for the single product problems equal 20.

For the multiple product case, we first generated several pairs of problems with identical parameters except that one had a setup cost equal to 0 and the other equal to 20. We found that there was no significant difference between the integrality gaps for any pair of problems and that, further, the algorithm generated the same cuts to tighten the linear program for most of the pairs. Therefore, we then generated problems with setup cost equal to 0 and report results only for this case.

For the multiple product version of the problem, *with only a single unit of demand for each product over the horizon*, it is also straightforward to show that $v(CSPL_1) = v(CSP)$. Thus, in this case, only P additional constraints are needed to close the integrality gap to 0. We report on some 5 product problems of this type in our computational results. In some of these instances, the system generated more than 5 cuts since the separation algorithm first generates all inequalities violated by the first product before checking for inequalities violated by the next product.

In our initial experiments, we also found that most of the cuts generated by the system to tighten the linear programming relaxation were of the following type

$$\sum_{t=1}^{j-1} w_{pt} + y_{pj} + z_{pj+1} + z_{pj+2} + \ldots.. + z_{pt_{pq}} \geq q \qquad (22)$$

with $t_{p,q-1} + 1 \leq j \leq t_{pq}$. Therefore, for all the test problems, we also computed the objective value of the linear programmimg relaxation with inequalities (22) included a priori, for all j and p, to determine the effectiveness of these inequalities in reducing the integrality gap.

Computational Results

We solved 18 single-product problems, 21 2-product problems and 15 5-product problems. Table 1 summarizes the results for a sample of these problems. In this table, v(LP) refers to the optimal value of the linear programming relaxation CSPL, v(LP1) refers to the optimal value of the linear programming relaxation with inequalities (22) included, v(LP2) to the optimal value of the linear program with all the violated facet inequalities included and v(IP) to the optimal value of the original problem. The results in this table, show that the facet inequalities are very effective in reducing the integrality gap for all the

Table 1

Computational Results for Selected Problems

No. of products	No. of periods	Total demand	Change over cost	LP value v(LP)	LP value with (5.1) v(LP1)	LP value after cuts v(LP2)	No. of cuts	Optimal IP value v(IP)	No. of bran-ches	(v(IP)-v(LP))/v(IP) %	(v(IP)-v(LP1))/v(IP) %	(v(IP)-v(LP2))/v(IP) %
1	10	3	200	246.7	393.3	400.0	6	410.0	2	39.8	4.1	2.4
1	10	7	100	520.0	540.0	540.0	3	540.0	0	3.7	0.0	0.0
1	10	7	200	591.4	640.0	640.0	3	640.0	0	7.6	0.0	0.0
1	15	5	200	543.8	705.0	710.0	8	720.0	7	24.5	2.1	1.4
1	15	10	50	981.7	1010.0	1010.0	4	1010.0	0	2.8	0.0	0.0
2	10	7	200	446.7	736.7	760.0	17	760.0	0	41.2	3.1	0.0
2	15	10	200	988.3	1435.0	1502.5	20	1530.0	5	35.4	6.2	1.8
2	15	15	200	1975.0	2316.7	2400.0	9	2450.0	1	19.4	5.4	2.0
2	20	20	50	2278.6	2404.2	2420.0	10	2450.0	2	7.0	1.9	1.2
2	20	20	100	2657.1	2908.3	2940.0	10	3000.0	3	11.4	3.1	2.0
2	20	20	200	3414.3	3916.7	3980.0	11	4100.0	2	16.7	4.5	2.9
5	10	5	100	395.0	700.0	700.0	6	700.0	0	43.6	0.0	0.0
5	15	10	200	1436.0	2100.0	2100.0	29	2100.0	0	31.6	0.0	0.0
5	15	15	100	1970.0	2250.0	2250.0	16	2250.0	0	12.4	0.0	0.0
5	15	15	200	2890.0	3450.0	3450.0	16	3450.0	0	16.2	0.0	0.0

instances considered; for a majority of the instances, the addition of these inequalities resulted in an integer solution and the system did not require branch and bound. Even when the facet inequalities were not sufficient to ensure an integer solution, they narrowed the gap considerably so that branch and bound could easily find an optimal integer solution. In all cases, the facet inequalities reduced the gap to less than 4% even though the original gap was over 40% in some cases. We further find that inequalities (22) are by themselves very effective in reducing the integrality gap in most cases.

Table 2 provides summary information concerning the average integrality gaps for all the 1-, 2- and 5-product test problems. A surprising conclusion emerged from these results: the cutting plane routine finds an integer solution for all the 5-product problems, whereas the algorithm requires branch and bound for some of the single-product and 2-product problems. Further, the addition of only inequalities (22) to the linear programming relaxation CSPL is sufficient to obtain an integer solution for all the 5-product problems. However, even for some of the single product and 2-product problems for which the addition of all violated inequalities results in an integer solution, the inclusion of inequalities (22) alone is not enough to obtain this solution. We do not have a good explanation for this result (However, see our comment just prior to the inequality (21).) Perhaps, the result is just a consequence of the particular problem instances generated.

Table 2

Average Integrality Gaps for the Scheduling Problems

No. of products	Average $\frac{v(IP)-v(LP)}{v(IP)}$ %	Average $\frac{v(IP)-v(LP1)}{v(IP)}$ %	Average $\frac{v(IP)-v(LP2)}{v(IP)}$ %
1	13.4	1.8	0.8
2	24.8	2.4	0.8
5	24.9	0.0	0.0

Comparing the results for problems that are similar except for the changeover costs, we find that though the integrality gap is larger the higher the changeover cost, in most instances the system generates the same cuts to tighten the linear program and reduces the integrality gap to approximately the same amount. Thus, the performance of the cutting plane procedure seems to be independent of the value of the changeover costs.

We examined the possibility of obtaining an upper bound on the optimal value for the problem by constructing a feasible solution from the fractional solution obtained at the end of the cutting plane routine. However, the structure of the fractional solutions makes it difficult to develop a simple heuristic that will provide a good upper bound. Moreover, since in most instances the algorithm require very few branches to find an optimal integer solution, we do not think it worthwhile to pursue this issue further.

Finally, note that for the smaller problem instances, it is quite possible that using branch and bound directly after solving the linear programming relaxation of the problem, with no additional constraints added, will be as effective a method as our cutting plane procedure. To provide some comparison between these two approaches, in Table 3 we provide the solution times for a few problems using (i) our cutting plane/branch and bound procedure described earlier, and (ii) branch and bound directly after solving the linear programming relaxation. The results in this table suggest that the cutting plane method does not offer much computational advantage for the smaller problems, but that it does significantly reduce the solution time for the larger problem

Table 3

Comparison of Solution Times for the Changeover Cost Scheduling Problem

No. of products	No. of periods	Total demand	Changeover cost	CPU time for cutting plane algorithm (sec.)	CPU time for branch & bound after lp (sec.)
1	10	3	200	12	17
1	15	5	200	37	54
2	10	7	200	44	75
5	15	10	200	783	5773

8. Conclusions

Our objective in this paper has been to study the polyhedral structure of a prototypical scheduling model in the production planning literature and to subsequently use the results to develop efficient solution methods. Our research was motivated by the following observations. First, even though this problem has been the focus of extensive study, available results from the literature have been effective in obtaining good solutions only for special cases. Second, as for any integer programming problem, the problem can be formulated in several different ways and results about the structure and properties of alternative formulations should provide useful insights about modeling. Finally, the success of cutting plane procedures using facet inequalities in other problem domains indicates that these methods can be very effective in solving large integer programming problems to optimality in reasonable computation times.

In this study, we proposed an integer programming formulation for the problem and identified a family of non-trivial facets of the convex hull of solutions to the problem. We also presented an efficient algorithm to solve the separation problem for this class of facets. This characterization of the polyhedral structure permitted us to implement and test a strong cutting plane procedure. Our computational results suggest that these facet inequalities are very effective in reducing the integrality gap -- for a large number of the problems tested, the addition of these inequalities resulted in an optimal integer solution, and even when the added constraints did not produce an integer solution, they considerably narrowed the integrality gap so that branch and bound could easily find an integer solution.

Though our computational results are very encouraging and indicate that these inequalities are effective in reducing the integrality gap, much more remains to be done. One potentially fruitful avenue for future investigation would be to obtain a more complete description of the convex hull of solutions to the problem. The multiple product problem is NP-complete and, hence, we do not expect to be able to obtain a compact characterization of the convex hull for this version of the problem (see, Grötschel, Lovasz and Schrijver, 1981 and Karp and Papadimitriou, 1982). On the other hand, the single product problem can be solved in polynomial time via dynamic programming and a complete characterization should be possible in this case. Results about how to tighten the general cut-set inequalities for the equivalent network design problem should yield additional facet inequalities for this problem and also provide insights that might prove to be useful for the general network design problem. Perhpas the most immediately useful follow on study, however, would be to perform further empirical tests and to implement, and as a result refine, the ideas presented in this paper on problems met in practice.

References

Balakrishnan, A., T.L. Magnanti, and R.T. Wong, (1987) A Dual-Ascent Procedure for Large Scale Uncapacitated Network Design, Working Paper OR 161-87, MIT Operations Research Center. To appear in *Operations Research.*

Barany, I., T.J. Van Roy and L.A. Wolsey (1984a) Uncapacitated Lot-Sizing: The Convex Hall of Solutions, *Mathematical Programming Study,* **22,** pp. 32-43.

Barany, I., T.J. Van Roy and L.A. Wolsey (1984b) Strong Formulations for Multi-Item Capacitated Lot-Sizing, *Management Science,* **30,** pp. 1255-1261.

Bomberger, E. E., (1966) A Dynamic Programming Approach to a Lot Size Scheduling Problem, *Management Science,* **12**, pp. 778-784.

Crowder, H., E. L. Johnson and M. W. Padberg, (1983) Solving Large-Scale Zero-One Linear Programming Problems, *Operations Research,* **31**, pp. 803-834.

Crowder, H. and M. W. Padberg, (1980) Solving Large-Scale Symmetric Travelling Salesman Problems to Optimality, *Management Science,* **26**, pp. 495-509.

Doll, C. L. and D. C. Whybark, (1973) An Iterative Procedure for the Single-Machine, Multi-Product Lot Scheduling Problem, *Management Science,* **20**, pp. 50-55.

Driscoll, W. C. and H. Emmons, (1977) Scheduling Production on One Machine with Changeover Costs, *AIIE Transactions,* **9**, pp. 388-395.

Elmaghraby, S. E., (1978) The Economic Lot Scheduling Problem (ELSP): Review and Extensions, *Management Science,* **24**, pp. 587-598.

Eppen, G. D. and R. K. Martin, (1985) Solving Multi-Item Capacitated Lot Sizing Problems Using Variable Redefinition, Working Paper, Graduate School of Business, University of Chicago. To appear in *Operations Research.*

Gascon, A. and R. C. Leachman, (1985) A Dynamic Programming Solution of the Dynamic, Multi-Item Single Machine Scheduling Problem, Document de travail 85-01, Faculte Des Sciences De L'Administration, Universite Laval, Canada.

Geoffrion, A. M., (1974) Lagrangean Relaxation for Integer Programming, *Mathematical Programming Study*, **2**, pp. 82-114.

Geoffrion, A. M. and G. W. Graves, (1976) Scheduling Parallel Production Lines with Changeover Costs: A Practical Application of a Quadratic Assignment / LP Approach, *Operations Research*, **24**, pp. 595-610.

Glassey, C. R., (1968) Minimum Change-over Scheduling of Several Products on One Machine, *Operations Research*, **16**, pp. 342-352.

Graves, S. C., (1980) The Multi-Product Production Cycling Problem, *AIIE Transactions*, **12**, pp. 233-240.

Grötschel, M., (1985) Polyhedral Combinatorics, in *Combinatorial Optimization: Annotated Bibliographies*, O'hEigeartaigh. M, J.K. Lenstra and A.H.G. Rinnooy Kan, eds, Wiley-Interscience, New York, pp. 1-10.

Grötschel, M., L. Lovasz and A. Schrijver, (1981) The Consequences of the Ellipsoid Method for Combinatorial Optimization, *Combinatorica*, **1**, pp. 169-198.

Haessler, R. W., (1979) An Improved Extended Basic Period Procedure for Solving the Economic Lot Scheduling Problem, *AIIE Transactions*, **11**, pp. 336-340.

Hodgson, T. J., (1970) Addendum to Stankard and Gupta's Note on Lot Size Scheduling, *Management Science*, **16**, pp. 514-517.

Hoffman, K. and M.W. Padberg, (1985) LP-Based Combinatorial Problem Solving, *Annals of Operations Research*, **4**, pp. 145-194.

Johnson, E. L., M. M. Kostreva and H. H. Suhl, (1985) Solving 0-1 Integer Programming Problems Arising from Large Scale Planning Models, *Operations Research*, **33**, pp. 803-820.

Karmarkar, U. S., S. Kekre and S. Kekre, (1987) The Dynamic Lotsizing Problem with Startup and Reservation Costs, *Operations Research*, **35**, pp. 389-398.

Karmarkar, U. S. and L. Schrage, (1985) The Deterministic Dynamic Product Cycling Problem, *Operations Research*, **33**, pp. 326-345.

Karp, R. M. and C. H. Papadimitriou, (1982) On Linear Characterization of Combinatorial Optimization, *SIAM Journal of Computing*, **11**, pp. 620-632.

Leachman, R. C. and A. Gascon, (1985) The Dynamic Cycle Lengths Heuristic for Scheduling Muti-Item, Single-Machine Production Systems, Part I, Document de travail 85-14, Faculte Des Sciences De L'Administration, Universite Laval, Canada.

Leachman, R. C. and A. Gascon, (1985) The Dynamic Cycle Lengths Heuristic for Scheduling Muti-Item, Single-Machine Production Systems, Part II, Document de travail 85-15, Faculte Des Sciences De L'Administration, Universite Laval, Canada.

Leung, J., T.L. Magnanti and R. Vachani, (1987) Facets and Algorithms for Capacitated Lot Sizing, Working Paper OR 171-87, Operations Research Center, MIT, Cambridge, MA.

Manne, A. S., (1958) Programming of Economic Lot Sizes, *Management Science*, **4**, pp. 115-135.

Marshall, P. W., W. J. Abernathy, J. G. Miller, R. P. Olsen, R. S. Rosenbloom and D. D. Wyckoff, (1975) *Operations Management: Text and Cases*, R.D. Irwin, Homewood, Illinois.

Mitsumori, S., (1972) Optimal Production Scheduling of Multicommodity in Flow Line, *IEEE Transactions, Systems, Man and Cybernetics*, SMC-2, pp. 486-493.

Orlin, J. B., (1986) Private Communication.

Pochet, Y., (1986) Valid Inequalities and Separation for Capacitated Economic Lot Sizing, Discussion Paper No. 8649, Center for Operations Research and Econometrics, University of Louvain, Louvain-La-Neuve, Belgium.

Schrage, L., (1982) The Multiproduct Lot Scheduling Problem, in *Deterministic and Stochastic Scheduling*, M.A.H. Dempster, J.K. Lenstra and A.H.G. Rinnooy Kan, eds., NATO Advanced Study Institutes Series, D. Reidel, Holland, pp. 233-244.

Stankard, M. F. and S. K. Gupta, (1969) A Note on Bomberger's Approach to Lot Size Scheduling: Heuristic Proposed, *Management Science*, **15**, pp 449-452.

Tenzer, N., (1969) Multi-Product Scheduling Problems with Time-Independent Setup Costs, Ph. D. Thesis, School of Engineering and Applied Science, Columbia University.

Vachani, R., (1986) Polyhedral Structure of Fixed Charge Production Planning and Scheduling Problems, Ph. D. Thesis, Sloan School of Management, MIT.

Vergin, R. C. and T. N. Lee, (1978) Scheduling Rules for the Multiple Product Single Machine System with Stochastic Demand, *INFOR*, **16**, pp. 64-73.