In many industries, production planning involves the allocation of various resources in the joint production of similar products. Inventory levels, labor decisions, and an economic choice of lot sizes are all influential in the planning process.

Formulated in terms of mathematical programming, the economic and mathematical facets of production planning are discussed. A feasible computation technique is suggested.

Fabrication and assembly operations

Part IV Linear programming in production planning

by B. P. Dzielinski

Industry has applied linear programming in a variety of ways. For instance, linear programming models have been used by individual firms in the metal working industries to plan the production rates of a single product and to determine the optimal assignment of jobs to machines.

This paper outlines a linear programming model that is applicable to a metal-working firm producing several different products jointly on a fixed collection of equipment. The decision-making model is concerned with economic lot sizes as well as inventory levels and work-force allocations. The lot-size decisions fix the quantity and timing of production for each product. Since quantities produced may not match demand schedules, inventory plans are also generated. The work-force decisions, which determine the labor force needed on a regular-time and on an overtime basis, provide the most economic levels of the plant labor force for producing the lot-size decisions.

One might ask, "Why use linear programming to study the question of economic lot sizes?" The answer is simply that, in metal working industries, production rate and lot size are two factors that can affect costs on a pronounced scale.[1,2] A linear programming model enables an individual firm to manipulate these factors and meet its demand requirements at minimum cost. In the selection of lot sizes, production economies can be effected by distributing setup costs over large quantities, but the indivisibilities of setup costs introduce discontinuities into the problem. On the other hand, lot sizes for the distinct products must observe plant capacity restrictions. A model applicable to lot-size decisions has been developed by Manne,[3] and used experimentally by Dzielinski, Baker, and Manne.[4]

Certain problems arise in the Manne formulation of the lot-size model when production must be planned for several thousand distinct products. First of all, it requires as many equations as products in the linear programming matrix. Secondly, a large number of variables are involved, because it is possible to specify many alternative production schedules for each product. Special techniques can be applied to overcome these problems by employing the Dantzig and Wolfe[5] decomposition principle to accomplish a reduction in the large number of equations. Also, the alternative production schedules can be generated, as they are needed, by means of the Whitin and Wagner[2] type of algorithm. This technique eliminates the task of computing and searching over a vast collection of existing schedules as required in the linear programming solution of the Manne economic lot-size model. These problems and their solutions are discussed in this part of the paper. The theoretical and programming aspects of the same problems have been treated by Dzielinski and Gomory.[6]

schedules

The Dzielinski, Baker, and Manne study introduced the concept of a *feasible production schedule*. If $g_{i\tau}$ denotes the delivery requirements of product i ($i = 1, \cdots, I$) in planning period τ ($\tau = 1, \cdots, T$), and $x_{ij\tau}$ is the amount of planned production of product i for the jth production schedule ($j = 1, \cdots, J$) in planning period τ, the following expressions must be satisfied for each i and j:

$$\sum_{\tau=1}^{t} x_{ij\tau} \geq \sum_{\tau=1}^{t} g_{i\tau} \qquad t = 1, \cdots, T-1 \tag{1}$$

and

$$\sum_{\tau=1}^{T} x_{ij\tau} = \sum_{\tau=1}^{T} g_{i\tau}, \tag{2}$$

where $x_{ij\tau} \geq 0$, $g_{i\tau} \geq 0$, and T is the total number of planning periods. Expressions 1 and 2 ensure a production schedule that at least satisfies the delivery requirements. Each product has 2^{T-1} such feasible schedules. The rules for generating these schedules have previously been discussed.[3,4] For example, we obtain the four feasible schedules of Table 1 if (1) for a given product, we generate production schedules which do not violate expressions 1 and 2, (2) we let $T = 3$, and (3) we have requirements g_1, g_2, g_3 (all greater than zero) for the first, second and third planning periods respectively. Other schedules would violate the conditions of Equations 1 and/or 2 or be "dominated" by these four schedules. In this method, the requirements making up the production quantities are never split into fractional parts; each $g_{i\tau}$ is scheduled in its entirety.

Table 1 Feasible production schedules

Schedule Number	*Amount produced for* $\tau = 1$	$\tau = 2$	$\tau = 3$
$j = 1$	$x = g_{i1} + g_{i2} + g_{i3}$	$x = 0$	$x = 0$
$j = 2$	$x = g_{i1} + g_{i2}$	$x = 0$	$x = g_{i3}$
$j = 3$	$x = g_{i1}$	$x = g_{i2} + g_{i3}$	$x = 0$
$j = 4$	$x = g_{i1}$	$x = g_{i2}$	$x = g_{i3}$

Let us now consider the production planning model of Dzielinski, Baker, and Manne, which assumes a plant that makes a variety of different products for each of which the delivery requirements $g_{i\tau}$ are assumed to be known. The plant can make many of these products simultaneously and, for reasons of economy, would like to arrange production so that the requirements are met and the load on the plant labor force in the various periods is smoothed. The most economic production schedules, when determined by linear programming, minimize on inventory holding costs, setup costs, regular payroll costs, overtime and shift premium payroll costs, and hiring and layoff costs—all subject to stated restrictions. Hence, the production planning model relates the feasible production schedules to labor requirements in the following manner. A labor coefficient $h_{ijk\tau}$ is calculated from the quantities $x_{ij\tau}$ in the feasible production schedules by the expression

$$h_{ijk\tau} = \begin{cases} 0 & \text{if } x_{ij\tau} = 0, \\ a_{ik} + b_{ik}x_{ij\tau} & \text{if } x_{ij\tau} > 0, \end{cases} \tag{3}$$

where a_{ik} and b_{ik} refer, respectively, to the setup time and the unit-production time of product i in facility (or labor group) k ($k = 1, \cdots, K$).

the cost

We now describe the production planning model, the *economic lot-size and optimal work-force planning model*. The objective is to minimize the cost equation

$$Z = \sum_{i=1}^{I} \left[\sum_{j} \theta_{ij} C_{ij} \right] + \sum_{k\tau} \left[\sum_{r} R_{kr\tau} W_{kr\tau} \right] + \sum_{k\tau} \left[\Gamma_{k\tau}^{+} W_{k\tau}^{+} + \Gamma_{k\tau}^{-} W_{k\tau}^{-} \right]. \tag{4.0}$$

The first expression, $\sum_j \theta_{ij} C_{ij}$, represents the inventory holding cost that is required to produce the ith product by the jth production schedule. The solution variable θ_{ij} is associated with the production schedule j. For each product, there is one θ_{ij} for each jth schedule, and each θ_{ij} has a corresponding cost coefficient C_{ij}. This coefficient represents the holding cost of inventory associated with each production schedule and is calculated as follows:

$$C_{ij} = \sum_{\tau=1}^{T} c_{i\tau}\, x_{ij\tau},$$

where $c_{i\tau}$ is a cost value that indicates the material cost of product i and is discounted with respect to period τ.

The expression $\sum_{k\tau} [\sum_r R_{kr\tau} W_{kr\tau}]$ represents the total cost of labor assigned to the labor solution variables $W_{kr\tau}$. Each $W_{kr\tau}$ represents the number of workers of labor group k (or assigned to facility k) and labor payment class r during period τ. Here, $r = 1$ is used for first shift, straight time, $r = 2$ for first shift, straight and overtime; $r = 3$ for second shift, straight time; $r = 4$ for second shift, straight and overtime; $r = 5$ for third shift, straight time; etc. The labor cost coefficient associated with a type kr worker in the τth period is denoted as $R_{kr\tau}$.

The last expression, $\sum_{k\tau} [\Gamma^+_{k\tau} W^+_{k\tau} + \Gamma^-_{k\tau} W^-_{k\tau}]$, represents the total cost of changing the size of the work force (number of workers) in any facility (labor group) k during period τ. The solution variables $W^+_{k\tau}$ and $W^-_{k\tau}$ represent the recommended hiring and layoff, expressed as number of workers in facility (labor group) k during period τ. Their respective cost coefficients are denoted as $\Gamma^+_{k\tau}$ and $\Gamma^-_{k\tau}$.

labor requirements

Minimization of the cost equation is subject to a set of associated side conditions. The first of these conditions is

$$\sum_{i=1}^{I} \left[\sum_j h_{ijk\tau} \theta_{ij} \right] - \sum_r H_{kr\tau} W_{kr\tau} \leq 0 \qquad \text{for all } k,\ \tau. \tag{4.1}$$

This is the equation of labor requirements in planning the production for I products. The expression in the bracket, the labor man-hours required, contains the labor coefficient $h_{ijk\tau}$ that is derived from each feasible production quantity $x_{ij\tau}$. Thus, the production schedules are represented as labor man-hour schedules. The total plant labor assigned is given by $\sum_r H_{kr} W_{kr\tau}$, in which H_{kr} denotes the number of hours a type kr worker works in a period.

convexity

In the production planning model, the alternative production schedules specified for each product are associated with the solution variable θ_{ij} for which the following conditions must be observed:

$$\left. \begin{aligned} &\sum_j \theta_{ij} = 1 \qquad \text{for all } i \\ &\text{and} \\ &\theta_{ij} \geq 0. \end{aligned} \right\} \tag{4.2}$$

The variable θ_{ij} requires some interpretation here. If the θ_{ij} are integers, Equations 4.2 assure that they can be only 0 or 1. For fixed i, only one θ_{ij} can be 1, the others must be 0. Thus, if all the

θ_{ij} are integers and satisfy 4.2, they can be interpreted as selecting, from the existing list of labor coefficients, for each product exactly one production schedule to be followed. However, according to Manne,[3] it is possible that the linear programming method produces proper positive fractions of the variable θ_{ij}. There are $I + m$ equations, in which m is determined by the values of K and T. By Equation 4.1, there must be at least one positive θ_{ij}. This accounts for I of the $I + m$ positive variables. Thus, there are at most m products for which more than one θ_{ij} is positive. Since only one θ_{ij} is positive whenever I is much larger than m, the number of m variables at proper fractional values is limited and can be treated by some appropriate rounding process.

labor balance

A so-called labor balance equation augments the production planning model:

$$W_{k\tau}^{-} - W_{k\tau}^{+} + \sum_{r} W_{kr\tau} = \sum_{r} W_{k,r,\tau-1} \qquad \text{for all } k,\ \tau. \tag{4.3}$$

This equation relates the size of work force from one period to the next. Different values of the work force variables $W_{k\tau}^{+}$ and $W_{k\tau}^{-}$ for each planning period τ indicate that the plant should take on more workers than currently available or should reduce its current labor force. However, hiring and dismissal costs must be considered when evaluating such results.

labor capacity

The production planning model was designed to assure that certain production plant capacities would not be violated. The following equations are used for this purpose:

$$\left.\begin{aligned} W_{k1\tau} + W_{k2\tau} &\leq M_k \quad \text{first shift} \\ W_{k3\tau} + W_{k4\tau} &\leq M_k \quad \text{second shift} \\ W_{k5\tau} + W_{k6\tau} &\leq M_k \quad \text{third shift} \end{aligned}\right\} \quad \text{for all } k,\ \tau, \tag{4.4}$$

where M_k denotes the plant capacity for facility (labor group) k, expressed in number of workers. For example, the first of these equations states that the sum of the number of workers assigned to straight time only ($W_{k1\tau}$) and to straight plus overtime ($W_{k2\tau}$), both in the first shift and in facility (labor group) k during period τ, may not exceed the value M_k. The other two equations refer in similar manner to the second and third shifts.

The following conditions are adjoined to Equations 4.0 through 4.4.

$$\left.\begin{aligned} W_{kr\tau} &\geq 0 \\ W_{k\tau}^{+} &\geq 0 \\ W_{k\tau}^{-} &\geq 0 \\ \theta_{ij} &\geq 0 \end{aligned}\right\} \quad \text{for all } k, r,\ \tau,\ i,\ j, \tag{4.5}$$

i.e., none of the solution variables may be assigned negative values.

The nature of the linear programming matrix, as described in Equations 4, is illustrated in Table 2. The main substance of the model lies in the labor requirement constraint Equation 4.1 and the constraint Equations 4.2.

The model discussed was tested on some data obtained from a real operating plant. The plant model, applied to the production of 35 distinct products, contained two facilities. It was decided to plan production for three periods on a 2-shift basis. The data were generated according to the definitions and rules of Equations 1 through 4. A standard linear programming code[7] for an IBM 704 was used to obtain the optimum feasible solution. An example of the output of this solution is shown in Table 3.

numerical example

Examination of this table shows, for instance, that variable θ_{14} (product 1, schedule 4) was in the optimal solution. The table indicates that production of this product occurred only in the second facility and took place in all three periods exactly as required. On the other hand, variable $\theta_{25,4}$ shows that the production for the requirements of period 3 was made with the requirements of period 2, and that product 25 required both facilities.

Examination of the work-force variables shows that a full complement of workers in facility 1 was allocated for all time periods and shifts; this value was 5, the capacity of this facility. For facility 2, we had a full complement of 117 workers on the first shift only; in fact, W_{221} (being in the optimal solution) shows that these workers were required to work overtime the first period, but not in the second or third periods. For shifts 2, only 3 workers were required in facility 2. The value of $W_{21}^{+} = 3$ indicates that 3 workers were hired at the start of period 1. This was necessary because no workers were available at the start of period 1 on the second shift of facility 2.

The disposal activities (slack variables) give interesting clues to the nature of the method. The numbers 281, 52, 0 refer only to facility 1. Even though 5 workers were allocated on each shift, not all of their time was needed for production work; for period 1, in fact, there were 281 hours of idle time. In period 2, there were only 52 hours of idle time, and in period 3, there were none. The method determined that it was less costly to pay the idle time than to dismiss first and rehire later on this facility when labor would be needed in the later periods.

The results on facility 2 are in reverse, since the numbers 0, 273, 946 indicate complete utilization of labor in the first period, 273 idle hours in the second period, and 946 idle hours in the third period. The method decided that the cost of this idle time was still less than dismissal cost for these workers. The values of 114 appearing in the lower right-hand corner of Table 3 indicate the number of additional workers that could be hired for facility 2.

Many earlier computer runs with this model[4] did not apply the special techniques used in this example.

model properties

When Equations 4 are converted into a linear programming matrix, they consist of $I + 5KT$ constraint rows and $(2^{T-1}) I + 8KT$ columns (slack and artificial columns not included). This formulation leads to several problems. If I, the number of distinct products, equals or exceeds several hundred, an equally high number of constraint rows of the matrix is generated, and

Table 3 Example of optimum feasible production schedules

	Product variables in the solution				Work-force variables in the solution														Disposal activities in the solution								
	θ_{14}	$\theta_{24} \cdots$	$\theta_{25,4} \cdots$	$\theta_{35,4}$	W_{111}	W_{221}	W_{112}	W_{212}	W_{113}	W_{213}	W_{131}	W_{241}	W_{132}	W_{232}	W_{133}	W_{233}	W_{21}^{+}	S_{11}	S_{21}	S_{12}	S_{22}	S_{13}	S_{23}	S_{21}	S_{22}	S_{23}
Cost Row	C_{14}	$C_{24} \cdots$	$C_{25,4} \cdots$	$C_{35,4}$	R_{111}	R_{221}	R_{112}	R_{212}	R_{113}	R_{213}	R_{131}	R_{241}	R_{132}	R_{232}	R_{133}	R_{233}	Γ_{21}^{+}	0	0	0	0	0	0	0	0	0
Prod. 1	1																									
Prod. 2		1																								
.		.																								
.		.																								
.		.																								
Prod. 25			1																							
.			.																							
.			.																							
.			.																							
Prod. 35				1																						
τ k																										
1 1	0	0	5	0	$5H_{11}$						$5H_{13}$							281								
1 2	828	0	359	89		$117H_{22}$						$3H_{24}$							0							
2 1	0	44	8	0			$5H_{11}$						$5H_{13}$							52						
2 2	629	0	570	274				$117H_{21}$						$3H_{23}$							273					
3 1	0	381	0	0					$5H_{11}$						$5H_{13}$							0				
3 2	629	0	0	274						$117H_{21}$						$3H_{23}$							946			
1 2																	3									
1 2																								114		
2 2																									114	
3 2																										114

Table 2 Product schedule variables and work-force variables

	Product schedule variables												
Variables	θ_{11}	θ_{12}	$\cdots$	θ_{1J}	θ_{21}	θ_{22}	$\cdots$	θ_{2J}	$\cdots$	θ_{I1}	θ_{I2}	$\cdots$	θ_{IJ}
Cost coefficients	C_{11}	C_{12}	$\cdots$	C_{1J}	C_{21}	C_{22}	$\cdots$	C_{2J}	$\cdots$	C_{I1}	C_{I2}	$\cdots$	C_{IJ}
Labor requirements (k_T rows)	h_{1111} h_{1112} $\vdots$ h_{11KT}	h_{1211} h_{1212} $\vdots$ h_{12KT}	$\cdots$ $\cdots$ $\cdots$	h_{1J11} h_{1J12} $\vdots$ h_{1JKT}	h_{2111} h_{2112} $\vdots$ h_{21KT}	h_{2211} h_{2212} $\vdots$ h_{22KT}	$\cdots$ $\cdots$ $\cdots$	h_{2J11} h_{2J12} $\vdots$ h_{2JKT}	$\cdots$	h_{I111} h_{I112} $\vdots$ h_{I1KT}	h_{I211} h_{I212} $\vdots$ h_{I2KT}	$\cdots$ $\cdots$ $\cdots$	h_{IJ11} h_{IJ12} $\vdots$ h_{IJKT}
Convexity constraints (I rows)*	1	1	$\cdots$	1									
					1	1	$\cdots$	1					
									$\ddots$				
										1	1	$\cdots$	1
Labor balance equations (k_T rows)*													
Labor capacity equations (k_T rows per shift)*													

*Note: Empty positions signify zeros.

the computation task can go beyond the capabilities of current computer programs. Furthermore, in many applications, the planning function may require T (the total number of planning periods) to be in the neighborhood of 10, and K (the number of distinct facilities) in the neighborhood of 25, generating $5KT$, or 1250, additional constraint rows. It is possible to overcome this problem by reducing the number of rows through approximations, forming the neighborhood of 10, we have $2^{10-1} = 512$ separate activities for each product, and if production is planned for 500 products, a total of 256,000 columns (activities) must be generated for the programming matrix. The simplex algorithm used to solve the problem of Equations 4 is faced with a very large computational task, since, on each iteration, it must determine from this large collection of activities the best one for improving the criterion function.

reducing the model size

The methods for solving the problem of the large number of rows and columns that can be generated from Equations 4 use two techniques: (1) the Dantzig and Wolfe Decomposition Principle[5] for overcoming the problem of the large number of constraint rows, and (2) the Whitin and Wagner[2] dynamic version of the economic lot-size formula for overcoming the large number of columns. Their applications to this problem are fully described by Dzielinski and Gomory.[6] In terms of the model, the methods of solution can be described in the following way:

1. The linear programming matrix is split into several parts. From Equation 4.1, a submatrix A_1 is obtained which consists of the labor coefficients. A second submatrix, A_2, can be obtained from Equations 4.2. A_2 is associated with A_1 and consists of the convex constraint coefficients.

$$A_1 = \begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1J} & C_{21} & \cdots & C_{IJ} \\ h_{1111} & h_{1211} & \cdots & h_{1J11} & h_{2111} & \cdots & h_{IJ11} \\ h_{1112} & h_{1212} & \cdots & h_{1J12} & h_{2112} & \cdots & h_{IJ12} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ h_{111T} & h_{121T} & \cdots & h_{1J1T} & h_{211T} & \cdots & h_{IJ1T} \\ h_{1121} & h_{1221} & \cdots & h_{1J21} & h_{2121} & \cdots & h_{IJ21} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ h_{11KT} & h_{12KT} & \cdots & h_{1JKT} & h_{21KT} & \cdots & h_{IJKT} \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 1 & 1 & \cdots & 1 & & & \\ & & & & 1 & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{bmatrix}$$

2. The combinations of A_1 and A_2 can be solved by themselves as a linear program, the problem being to maximize $\pi \cdot A_1\theta$ subject to $A_2\theta = d$, where π refers to the price vector in an ordinary linear programming problem, and d represents the right side of Equations 4.2. Although this problem is almost as large as the original problem, it can be solved much more easily. If we denote by A_{ij} the column of A_1 corresponding to the variable θ_{ij}, we see that the

above expression max $\pi \cdot A_1\theta$ splits up into a series of separate problems, one for each product i, and each of the form

$$\max\ (\pi\cdot A_{i1}\ ,\ \pi\cdot A_{i2}\ ,\ \cdots\ ,\ \pi\cdot A_{iN_i})\cdot(\theta_{i1}\ ,\ \cdots\ ,\ \theta_{iN_i}), \tag{5}$$

subject to

$$\sum_{j=1}^{N_i} \theta_{ij} = 1.$$

For the solution, we simply set θ_{ij} equal to 1 for the j value for which the scalar product $\pi\cdot A_{ij}$ is greatest, and set all other θ_{ij}'s to 0.

Thus far, only a portion of the entire problem of Equations 4 has been solved here, namely, the economic lot-size decisions for each product. The next step is to determine if the economic lot-size decisions, when combined for the I products, are feasible with regard to plant capacities and, if so, what should be the most economic labor allocations to produce the given lot sizes for an overall feasible or optimally feasible solution.

3. A so-called "master problem" is formed and solved as an ordinary linear programming problem. The independent economic lot-size solutions obtained at this stage for each product i are combined to give, for the whole problem, a solution which not only includes Equations 4.1, 4.3, and 4.4, but also a single additional equation that introduces the combined product solutions into the overall solution by a single variable with an upper bound of 1.

4. From the master-problem solution, we derive a new price vector π which includes a price component for each equation. This component represents the marginal value of each facility (labor group) and time period.

5. We apply these prices to the variables in Equation 5, the so-called independent problems for each product i, and repeat the previous steps, beginning with step 2.

At each stage, the new solutions from step 2 are combined and added as a variable to the master problem. The process terminates when the solutions to the independent problems are the same in successive iterations.

In the computation of Equation 5, we investigate scalar products as numerous as schedules on every iteration of the optimization process. Fortunately, finding the largest $\pi \cdot A_{ij}$ for fixed i can be done by a recursive dynamic programming calculation of the Whitin and Wagner type.[2] The problem is to find for each i the "dominant" schedule j that minimizes

$$\begin{aligned}\pi\cdot A_{ij} &= \sum_{k\tau} -\ \bar{\pi}_{k\tau}(a_{ik}\delta(x) + b_{ik}x_{ij\tau}) + C_{ij} \\ &= \sum_{\tau} \{A_{i\tau}\delta(x) + B_{i\tau}x_{ij\tau}\},\end{aligned} \tag{6}$$

where $\delta(x) = 0$ for $x = 0$, $\delta(x) = 1$ for $x \neq 0$, and

$$A_{i\tau} = \sum_{k} -\ \bar{\pi}_{k\tau}a_{ik}$$

$$B_{i\tau} = \sum_k - \bar{\pi}_{k\tau} b_{ik} + c_{i\tau}.$$

The right side of Equation 6 is merely an expanded description of $\pi \cdot A_{ij}$, where the price vector π contains a component for each constraint equation in the programming matrix. These components give an evaluation of the economic significance of the equation. In particular, the $\bar{\pi}_{k\tau}$ values represent the economic value of the labor requirement constraints (i.e., Equation 4.1), and for our purposes are interpreted as the unit labor cost for the labor coefficients in each production schedule A_{ij}. Thus, by rearranging the terms, we obtain two values, $A_{i\tau}$ and $B_{i\tau}$, for each period τ.

In viewing Equation 6 as an economic lot-size problem, $A_{i\tau}$ becomes the setup cost and $B_{i\tau}$ the marginal cost of production. To minimize the equation, we introduce the function $C_{i\tau}(y)$. This function is the minimum cost of filling all requirements $g_{i\tau}$ up to and including the τth period and having an amount y of extra production on hand at the end of the τth period. $C_{i\tau}(y)$ can be obtained recursively from

$$C_{i1}(y) = A_{i1}\delta(y + g_{i1}) + B_{i1}(y + g_{i1}) \tag{7.1}$$

$$C_{i\tau}(y) = \min \{A_{i\tau}\delta(z) + B_{i\tau}(z) + C_{i,\tau-1}(y + g_{i\tau} - z)\} \quad 0 \le z \le y + g_{i\tau} \quad \text{and} \quad \tau > 1 \tag{7.2}$$

Thus, to obtain $C_{iT}(0)$, the minimum cost of filling all orders, it is only necessary to compare values corresponding to $z = 0$ and $z = y + g_{i\tau}$ approximately $T(T + 1)/2$ times. Backtracking to obtain the $x_{ij\tau}$ that gave this cost is only a very small additional calculation which is substituted for evaluating 2^{T-1} scalar products for each product i.

concluding remarks

By a Dantzig and Wolfe decomposition, followed by an application of a column generating technique, a problem originally calling for simplex computations on an $(I + 5KT)(I + 5KT)$ matrix (and for the investigation of scalar products with columns as numerous as schedules) is reduced to a problem calling for simplex computations on a $(5KT + 1)(5KT + 1)$ matrix and for a string of I associated dynamic programming calculations.

The above formulations were coded and tested in an IBM 7090 experimental computer program. The code consists of a simplex algorithm needed to solve the master problem, and a dynamic programming algorithm to solve the economic lot-size problems. The code is designed to allow the simplex algorithm to evaluate an ever-increasing number of columns. These additional columns, which we denote as production plan vectors, are systematically generated as needed for the master problem through the use of recursive Equation 7.2. The elements of the production plan vectors are the combined labor hours of the economic lot-size solutions for the individual products. The same capability is now being implemented as the Discrete Production Resource Allocator (DPRA) in the 7040/44 Linear Programming System III.[8]

Table 4 presents a summary of some test problems solved by the experimental computer program. Part I of the table shows the effect of different values on the size of the linear programming matrix. Part II shows the magnitudes of the rows and columns if the problem of Equations 4 were set up as an ordinary linear programming problem. Therefore, the simplex algorithm would be required to iterate on a basis of close to 500 rows. Part III shows the magnitude of the rows and columns of the master problem and the I associated dynamic programming calculations when problems are solved by the experimental program. In part IV, computational results are given for problems labelled A, B, and C.

Some important characteristics are indicated by Table 4. Specifically, calculations are performed on a much smaller scale than is indicated by only an application of the decomposition prin-

Table 4 Sample problem parameters and solution characteristics

		A	B	C
I. *Parameters that determine size of problems*				
1. I	number of products	428	428	428
2. K	number of facilities	2	2	2
3. T	number of periods	3	5	7
4. S	number of shifts	2	2	2
5. $(S + 2)KT + 2$	work force constraint rows	26	42	58
6. $2KT + 3$	work force columns	36	60	84
7. $3KT$	slack columns	18	30	42
8. $(2^{T-1})I$	total production schedules	1712	6784	27392
II. *Size of problems as ordinary LP Problems*				
9. $(1) + (5)$	number of rows	454	466	486
10. $(6) + (7) + (8)$	number of columns	1766	6874	27518
III. *Size of problems decomposed with dynamic programming*				
11. (5)	number of rows	26	42	58
12. $(6) + (7)$	number of columns*	54	60	84
13. $(1)\ T(T + 1)/2$	number of calculations to solve the I dynamic programming problems	2568	6420	11984
IV. *Computations on the test decomposed problems with dynamic programming*				
14. total time to compute optimal solution**		7.52	11.94	20.76
15. total phase two iterations		22	35	52
16. time to create a production plan (an average in minutes		0.352	0.460	0.680
17. number of production plans created		20	24	26
18. time between iterations, not including time to create a production plan (average in min.)		0.009	0.01	0.047
19. number of production plans in the optimal solution		6	8	11

* The column numbers represent the number available at the start of the computations of the master problem. Column additions are made to the problem during the computations; for the sample problem, these magnitudes are shown in IV, 17.

** This time does not include machine time for setting up the problem and outputting the optimal solution.

ciple. The advantages of the dynamic programming algorithm are especially significant for Problem C: fewer production schedules need be evaluted for the increased number of planning periods.

This approach considers all possibilities and has the advantage that it is only necessary to read product requirements, material costs, setup times, and process times into the computer. The machine generates the feasible production schedules as they are required. It is not necessary to create, maintain, or update a large file of feasible production schedules.

CITED REFERENCES

1. S. B. Alpert, "Economics of scale in the metal removal industry," *Journal of Industrial Economics,* **7**, No. 3, 175–181 (July 1959).
2. H. M. Wagner and T. M. Whitin, "A dynamic version of the economic lot size model," *Management Science* **5**, No. 1, 89–96 (1958).
3. A. S. Manne, "Programming of economic lot sizes," *Management Science* **4**, No. 2, 115–135 (1958).
4. B. P. Dzielinski, C. T. Baker, and A. S. Manne, "Simulation tests of lot size programming," *Management Sciences* **9**, No. 2, 229–258 (January 1963).
5. G. B. Dantzig and P. Wolfe, "Decomposition principle for linear programs," *Operations Research* **8**, No. 1, 101–111 (1960).
6. B. P. Dzielinski and R. E. Gomory, "Optimal programming of lot size inventories and labor allocations," IBM ASDD Technical Report TR 17-120. International Business Machines Corporation, Yorktown Heights, New York.
7. W. Orchard-Hays and J. R. Hills, "SCROL, a comprehensive operating system for linear programming on the IBM 704," CEIR, Inc., 1200 Jefferson Davis Highway, Arlington 2, Va.
8. "7040/7044 linear programming system III, application description," H20-0119-0, International Business Machines Corporation, Data Processing Division, White Plains, New York.

Symbol key

i product index $(1, \cdots, I)$
j production schedule index $(1, \cdots, J)$
k facility (or labor group) index $(1, \cdots, K)$
r labor payment class
τ planning period index $(1, \cdots, T)$
g delivery requirements
x planned production
a setup time
b unit-production time
h labor coefficient associated with variable x
H number of worker hours
R cost coefficient associated with variable W
Γ cost coefficient associated with variable W
C cost coefficient associated with variable θ
M plant capacity, expressed in number of workers
W number of workers
θ variable associated with product schedules
π price vector