



Concurrent consideration of product design, process planning and production planning activities

Gajendra Kumar Adil

To cite this article: Gajendra Kumar Adil (1998) Concurrent consideration of product design, process planning and production planning activities, *Production Planning & Control*, 9:2, 167-175, DOI: [10.1080/095372898234389](https://doi.org/10.1080/095372898234389)

To link to this article: <http://dx.doi.org/10.1080/095372898234389>



Published online: 15 Nov 2010.



Submit your article to this journal [↗](#)



Article views: 42



View related articles [↗](#)

Concurrent consideration of product design, process planning and production planning activities

GAJENDRA KUMAR ADIL

Keywords concurrent engineering, integration of manufacturing functions, production planning, product design, process planning

Abstract. In manufacturing engineering, product design, process planning and production planning activities are often considered independently. However, in order to effectively respond to changes in business situations, such as changes in demand forecast, product mix and technology, it is desirable to consider them concurrently. For this purpose, a large-scale linear programming model has been developed. The model considers minimization of the sum of processing cost, late shipment cost and inventory holding cost as the objective, and concurrently selects product designs, and generates process plans and production plans. The number of columns in the formulation can be large and, hence, an efficient column generation scheme is developed to solve the model. The model and solution procedure are illustrated with examples.

1. Introduction

Concurrency and integration of various manufacturing functions are important for any manufacturing organization to effectively respond to the current dynamic business situations (Arcelus and Srinivasan 1995). This has led to the emergence of concurrent engineering (CE) and computer integrated manufacturing (CIM) concepts,

and rapid progress in CE and CIM tools and technologies. This paper considers three important manufacturing functions simultaneously: product design, process planning, and production planning.

Today, product diversification rather than product standardization is the market need. Custom-made products, with a great many variations in design are common. To have many product variations helps to satisfy customers. In addition, this can provide a manufacturing organization with flexibility in resource management. With judicious selection of design (product) mix, it is possible to ensure efficient use of resources (such as good load balance and high utilization) because the availability of a large number of designs gives flexibility in resource selection.

Process planning is a function which associates each design feature (or group of design features) to manufacturing resource(s) to be used for its creation. There is no unique way in which a product with a given design can be manufactured. This leads to the existence of many process plans corresponding to a given product design. Many researchers (Horvath *et al.* 1996) have worked on the automatic generation of process plans for the product design for a class of products.



Author: Gajendra Kumar Adil, Department of Industrial Engineering, Bilkent University, 06533 Bilkent, Ankara, Turkey; Address for correspondence: 133-99 Dalhousie Drive, Winnipeg, Manitoba, Canada, R3T 3M2.

GAJENDRA KUMAR ADIL worked as an assistant professor at Bilkent University from September 1995 to August 1996 and this research was carried out during that period. Currently, he is working as a consultant for Bristol Aerospace Limited, Winnipeg, Canada. He holds a PhD (1994) in Industrial Engineering from the University of Manitoba, an MSc (1990) in Industrial Engineering from the University of Regina, an M.Tech (1987) in Mechanical Engineering from the Indian Institute of Technology, Kanpur and a B.Engg (1985) in Mechanical Engineering from Ravishankar University, Raipur, India. His research interests are in cellular manufacturing, production planning and concurrent engineering.

Production planning decisions may be many. One important medium- and long-range production planning decision is to determine the number of parts to be produced during each planning period. This decision depends to a great extent on (i) demand forecast, and (ii) resource availability. How a product is designed can affect demand. The design and process plan selections determine the resource requirements. Thus, product design, process planning and production planning influence each other, and concurrent consideration of these three should help a company in optimizing its operation. (We will explain some of the iterations with examples in Section 4.)

Some authors have considered the relationship between design and process planning (Dewhurst and Boothroyd 1988, Horvath *et al.* 1996), and a few (ElMaraghy and ElMaraghy 1992, Larsen and Clausen 1992, Huang *et al.* 1995) have simultaneously considered process planning and production planning activities. The objective of this paper is to develop a procedure to achieve simultaneous design selection, process plan generation and production planning. We develop a model for concurrent selection of product design; generation of an optimal process plan by implicitly enumerating different options; and development of production plans.

In Section 2, a mathematical formulation of the problem is presented. The model can result in a large number of explicit columns, therefore, in Section 3, we develop a solution procedure based on a column generation scheme to solve large-size problems efficiently. The model and the importance of concurrent design selection, process plan generation and production planning decisions are illustrated in Section 4. Computational experience with the column generation scheme is presented in Section 5 and in Section 6 we discuss a possible variation of the model. Finally, conclusions are presented in Section 7.

2. Mathematical formulation

We shall introduce notation and terms before presenting the mathematical model

2.1. Notation

Indices

p	product type ($p = 1, 2, \dots, P$)
d	design (or variation) of product ($d = 1, 2, \dots, D_p$)
f	design feature ($f = 1, 2, \dots, F_{pd}$)
k	process plan ($k = 1, 2, \dots, K_{pd}$)
r	resource type ($r = 1, 2, \dots, R$)

t, t' time period in planning horizon ($t, t' = 1, 2, \dots, T$).

Parameters

B_{rt}	total processing time available on resource type r in time period t
$C_{rj}(pd)$	cost incurred in using resource r to create feature f specified in design d of product type p
D_{pt}	forecast production quantity (or demand) of product type p in planning period t
$h_{it}(p)$	inventory holding cost for each unit of product type p when the product is produced in period t to meet the demand in period t' ($t' > t$)
$h_{it'}(p)$	late shipment cost for each unit of product type p when the product is produced and supplied in period t against the demand in period t' ($t' < t$)
R_{pd}	number of resources in the cost (time) matrix corresponding to design d of product p
$T_{rj}(pd)$	time required on resource r in creating feature f specified in design d of product type p .

Decision variables

$$A_{rj}(kpdtt') = \begin{cases} 1 & \text{if, product type } p \text{ is produced in design } \\ & d \text{ in time period } t \text{ to meet the} \\ & \text{customer's demand in period } t', \\ & \text{using process plan } k \text{ that requires} \\ & \text{resource } r \text{ to create feature } f \\ 0 & \text{otherwise} \end{cases}$$

$$X(kpdtt') = \text{quantity of product type } p \text{ produced in} \\ \text{design } d \text{ using process plan } k \text{ in period } t \text{ to} \\ \text{meet the customer's demand in period } t'$$

2.2. Terms

We now explain three terms: product design, process planning and production planning, and relevant notation used in this paper.

2.2.1. Product design

The design department comes up with many variations of a given product p and provides a design specification for each variation. Let there be D_p designs (or variations) of product p . Each design specifies a number of features, f ($f = 1, 2, \dots, F_{pd}$) required to manufacture the product. Each design feature can be created using one of the alternative resources. However, the time and cost incurred may vary depending upon the resource selected. Let the time required and cost incurred on the selected

Table 1. Typical processing cost and time matrices

	(a) $T_{rf}(pd)$ for a given (pd)			(b) $C_{rf}(pd)$ for a given (pd)			
	Feature, f			Feature, f			
	1	2	3	1	2	3	
Resource, r	1	40	20	—	20	30	—
	4	—	—	10	—	—	25
	6	10	60	—	30	40	—

Note: ‘—’ indicates that the resource is not capable of producing that feature.

resource r to process feature f specified in design (or variation) d of product p be $T_{rf}(pd)$ and $C_{rf}(pd)$, respectively. They can be represented by matrices as shown in Table 1.

The following information can be extracted from the time and cost matrices in Table 1. Design (variation) d of product p specifies three design features. Each feature can be produced by one of the three resource types 1, 4 and 6 with processing time and cost indicated. In the matrices, the sign ‘—’ (for example, corresponding to $f = 3$ and $r = 1$) indicates that the resource is not capable of producing the corresponding feature. Sign ‘—’ can also be replaced by infinity (or a large number).

2.2.2. Process plans

Production according to the specifications outlined in a design (product variation) requires generation of a process plan. This can be obtained by choosing a feasible resource for each design feature. We can represent a process plan by a set containing all the design features and a feasible resource selected to produce each feature. For a given design there can be many feasible process plans. For example if the data given in Table 1 are considered, four feasible process plans are possible as shown in Table 2.

Table 2. Enumeration of process plans for data presented in Table 1.

Process plan, k	Representation of process plan [selected (feature, resource) combinations]	Total time (cost)
1	$\{(1, 1), (2, 1), (3, 4)\}$	70 (75)
2	$\{(1, 1), (2, 6), (3, 4)\}$	110 (85)
3	$\{(1, 6), (2, 1), (3, 4)\}$	40 (85)
4	$\{(1, 6), (2, 6), (3, 4)\}$	80 (95)

Note. The infeasible process plans represented with time and cost as infinity are not listed in the table. If we include these also then there will be a total of 27 process plans.

Note that infeasible process plans (represented with time and cost as infinity) are not listed in the table. If these are included there will be a total of 27 process plans. This can be calculated as the number of resources in the matrix to the power number of features = $R_{pd}^{F_{pd}} = 3^3$.

2.2.3. Production plans

The production planning decision involves determination of the quantity of each product type to be produced in each planning period given the amount of time available on resources and the demand forecast for the products. We may not always be able to produce just the demanded quantity in each time period because of constraints on resource availability or economic considerations. We may have situations when in some period we may produce more (or less) than the quantity demanded. If the demand is not met, companies lose current (or even future) business opportunities. Sometimes the company may not be able to ship the products late, after paying an agreed penalty to the customer. We define $h_{lr}(p)$ to be the late shipment cost for each unit of product type p when the product is produced and supplied in period t against the demand in period t' ($t > t'$). If late shipment is not possible, a very high cost value can be allocated in the model to disallow it. Sometimes, excess production capacity may be available in a given period. In that case a product can be produced in excess quantity to meet future demands. This results in inventory holding cost. We define, $h_{ur}(p)$ to be the inventory holding cost for each unit of product type p when the product is produced in period t to meet the demand in period t' ($t < t'$). We have been able to use the same variable $h_{ur}(p)$ to represent both the inventory cost and late shipment cost because for any given situation both cannot occur simultaneously.

We can generate different production and process plans to produce a given product p in design d . Let us define binary variables $A_{rf}(kpdtt')$ to represent all such options. $A_{rf}(kpdtt')$ takes the value 1 if all of the following conditions hold true: product type p in design d is produced in period t to meet the demand in period t' and uses process plan k in which resource r is selected to create the feature f . If any of the conditions is not true, the variable takes the value 0.

2.3. Mathematical model

Having defined the notation and explained the terms, we can now formulate a large-scale linear programming model as follows.

$$\begin{aligned} \text{Minimize } z = & \sum_r \sum_{p'} \sum_p \sum_d \sum_k \sum_j \sum_r A_{rj}(k p d t t') \\ & C_{rj}(p d) X(k p d t t') \\ & + \sum_r \sum_{p'} \sum_p \sum_d \sum_k \sum_r h_{it'}(p) X(k p d t t') \\ & + \sum_r \sum_{p'} \sum_p \sum_d \sum_k \sum_r h_{it'}(p) X(k p d t t') \end{aligned}$$

or

P0:

$$\begin{aligned} \text{Minimize } z = & \sum_r \sum_{p'} \sum_p \sum_d \sum_k \sum_j \sum_r A_{rj}(k p d t t') \\ & C_{rj}(p d) X(k p d t t') \\ & + \sum_r \sum_{p'} \sum_p \sum_d \sum_k \sum_r h_{it'}(p) X(k p d t t') \quad (1) \end{aligned}$$

Subject to:

$$\sum_r \sum_d \sum_k X(k p d t t') \geq D_{pt'} \quad \forall (p, t') \quad (2)$$

$$\begin{aligned} \sum_r \sum_p \sum_d \sum_k \sum_j A_{rj}(k p d t t') T_{rj}(p d) X(k p d t t') \leq B_{rt} \\ \forall (r, t) \quad (3) \end{aligned}$$

$$X(k p d t t') \geq 0 \quad \forall (k, p, d, t, t') \quad (4)$$

In the above formulation, objective function (1) is the sum of processing cost, late shipment cost and inventory holding cost. Constraints (2) ensure that the demand for all products are met and constraints (3) enforce the capacity limitation on resources. Constraints (4) impose non-negativity restrictions on production quantities.

3. Solution procedure

Model P0 is a large-scale linear programming model. The number of process plans and the resulting number of columns (if we explicitly consider these columns) in the model can be large. For instance, consider a design in which the number of design features is 10 and the number of resources to produce one or more features is five. The number of possible process plans for this design will be $5^{10} = 9765625$. Further, let there be 20 products, each of which is to be produced in five designs during five planning periods. The resulting number of explicit columns $[X(k p d t t')]$, would be equal to $9765625 \times 20 \times 5 \times 5 \times 5$ which is of the order 2.4×10^{10} . In the simplex procedure, pricing all the variables in order to determine the entering column is quite expensive. Therefore, we propose a column generation scheme to implicitly enu-

merate them. For the above situation, the proposed scheme will require pricing of only $1 \times 20 \times 5 \times 5 \times 5 = 2.5 \times 10^3$ variables because it does not require enumeration over process plan index k . The most promising process plan is obtained using the column generation scheme described below. A similar scheme is reported by Rajamani *et al.* (1996) in the context of cell design.

The structure of the model P0 is such that the generation of an X -column in each simplex iteration requires solving a semi-assignment problem. This can be solved using a simple approach. To explain this, let us define simplex multipliers corresponding to constraints (2) as $U_{pt'}$ (for $p = 1, 2, \dots, P$; $t' = 1, 2, \dots, T$) and that corresponding to constraints (3) as V_{rt} (for $r = 1, 2, \dots, R$; and $t = 1, 2, \dots, T$). A non-basic variable, $X(k p d t t')$ having reduced cost less than zero can improve the objective value. We need to check if the following condition is met for the corresponding column to enter,

$$\begin{aligned} \sum_j \sum_r A_{rj}(k p d t t') C_{rj}(p d) + h_{it'}(p) - U_{pt'} \\ - \sum_j \sum_r A_{rj}(k p d t t') T_{rj}(p d) \cdot V_{rt} < 0 \end{aligned}$$

or

$$\begin{aligned} \sum_j \sum_r A_{rj}(k p d t t') [C_{rj}(p d) - T_{rj}(p d) \cdot V_{rt}] \\ + h_{it'}(p) - U_{pt'} < 0 \quad (5) \end{aligned}$$

Since the range of values a process plan index k takes can be quite high we do not explicitly enumerate and price each of them. Instead, we generate the most promising process plan k^* , that gives the lowest value for the summation term in expression (5). This gives the following subproblem.

SP:

$$\begin{aligned} \text{Minimize } U(p, d) = \sum_r \sum_{p'} [C_{rj}(p d) - T_{rj}(p d) \cdot V_{rt}] \\ \times A_{rj}(k^* p d t t') \end{aligned}$$

Subject to:

$$\sum_r A_{rj}(k^* p d t t') = 1 \quad \forall j$$

$$A_{rj}(k^* p d t t') \in \{0, 1\} \quad \forall (r, j)$$

Note that the values of variables $A_{rj}(k^* p d t t')$ or the resource-feature combinations obtained by solving SP define the most promising process plan k^* . Further, the objective function coefficient in SP depends only on part p and design d . Thus this problem needs to be solved for each (p, d) combination.

Table 3. Data for example (case 1)

Number of product types, $P = 2$
 Number of designs, $D_1 = D_2 = 1$
 Number of planning periods, $T = 2$

Demands (D_{pt}) for products

Product type, p	Time period, t	
	1	2
1	10	20
2	20	10

Processing costs and times

Product 1		Product 2				
$T_{rf}(1,1)[C_{rf}(1,1)]$		$T_{rf}(2,1)[C_{rf}(2,1)]$				
	f		f			
r	1	2	r	1	2	3
1	10 [5]	—	2	—	10 [10]	—
4	—	50 [10]	3	10 [15]	—	—
			4	—	—	20 [10]

Time (B_{rt}) available on resources

Resource r	Planning period, t	
	1	2
1	200	200
2	500	500
3	300	300
4	600	500

Table 4. Capacity requirements on each resource in example (Case 1).

Resource	Planning period, t	
	1	2
1	100	200
2	200	100
3	200	100
4	900	300

Problem SP is separable by f . The resulting problem for each f can be solved by inspection, i.e. select the variable with the least value of its coefficients underlined in the objective function. Let the optimal objective value of SP by $U^*(p, d)$ for a given (p, d) . We select a design d^* that gives the minimum value of $U^*(p, d)$. We store the process plan for this design, i.e. store the $A_{rf}(k^*, p, d^*, t, t')$ matrix.

Now we need to determine the best values of periods t^* , t'^* for given part p . Using the above developments,

expression (5) reduces to

$$\text{or } F(p) < 0$$

where

$$F(p) = \text{minimum}_d [U^*(pd)] + \text{minimum}_{t'} \left[\text{minimum}_t \{h_{rt}(pp) - U_{pt'}\} \right]$$

Column $X(k^*, p, d^*, t^*, t'^*)$ offers the maximum cost improvement potential for part p . All such variables $X(k^*, p, d^*, t^*, t'^*)$ which have a negative value for $F(p)$ can enter. From this we select the column which gives the most negative value, i.e. we enumerate over all parts p and pick the column $X(k^*, p, d^*, t^*, t'^*)$ with the most negative value for $F(p)$. In other words,

$$F(p^*) = \text{minimum}_p \{F(p)\}$$

The column generation scheme thus described is illustrated with a numerical example in the Appendix.

A two-phase simplex method is used here to solve the linear programming problem. In phase I, initially, artificial variables of constraints (2) appear in the basis with a very high cost coefficient, M ; and in constraints (3) all slack variables appear with cost coefficient zero. The simplex algorithm stops when the best X -column found has a value of $F(p^*) \geq 0$ (i.e. no X -column can enter); $U_{pt'} > 0$ [i.e. no excess variable corresponding to constraints (2) can enter]; and $v_{rt} < 0$ [i.e. no slack variable corresponding to constraints (3) can enter]. The simplex algorithm combined with the column generation scheme proceeds as follows:

- Step 0. Initialize basis and solution. Initialize iteration count as 0.
- Step 1. Increase the simplex iteration count.
- Step 2. Find the entering variable (column):
 - (a) Check if an X column can enter using the column generation scheme described earlier in this section. If this happens then go to step 3.
 - (b) Check if an excess variable in constraints (2) can enter. If this happens then go to step 3.
 - (c) Check if a slack variable in constraint (3) can enter. If this happens then go to step 3.
 If algorithm is still in phase I, then the problem is unfeasible, or the optimality condition has been reached. Go to step 4.
- Step 3. Find the departing variable, update the basis, find the new solution for the current basis, and go to step 1.
- Step 4. Print the results and terminate the algorithm.

4. Illustrative examples

In this section we illustrate the importance of concurrent consideration of design selection, process planning and production planning activities with some examples. Out of the many possible trade-offs we shall consider only a few of them to bring out the salient points. Also, short examples are adequate for this purpose. Computation experience with large-size problems will be presented in Section 5.

Example (Case 1). Let us first consider only one design and one process plan for each product that corresponds to the minimum processing cost. Consider the data given in Table 3.

One design and one process plan are specified for each product. The processing costs are 15 and 35 for products 1 and 2, respectively.

The capacity (time) requirement of each type of resource can easily be computed from the demand and routing information. This is summarized in Table 4.

If we compare the available time (Table 3) with the required processing time (Table 4) on resources, it is obvious that resource 4 does not have enough capacity available to meet the demand completely, while resources 1, 2 and 3 have excess capacities. It can be observed that demand cannot be met even with late shipment and early production. Further, this gives rise to situations where some resources are idle while others are overloaded.

Example (Case 2). Now let us consider that we can produce a different variation (design) of product type 1 which is acceptable to the customer. However, this incurs a higher processing cost (20 per unit as compared to 15 per unit using the original design). In addition, let there be an alternative resource 4 (along with to resource 3), which can create feature 1 for product 2. If this option is used the process cost increases from 35 to 40 per unit for product 2. The revised time and cost matrices, and the late shipment and inventory holding cost $[h_{it}(p)]$ (assumed to be the same for both products) are shown in Table 5. All other data remain the same for case 1.

We solved model P0 with the revised data and the process plans obtained are shown in Table 6.

Here we notice that the lowest cost design (design 1 of product 1) and the lowest cost process plan (plan 2 of product 2) were not selected to meet all the demand. This could not be done because there was not enough capacity on resources, as we observed in case 1. Also, with additional options (in process plans and design variation), a smoother workload distribution can result (as observed from Table 7).

Clearly this result demonstrates that there are advantages to considering more than one design and more than one process plan wherever available, even though choos-

Table 5. Time and cost data for example (Case 2)

Product 1							
$R_{rj}(1,1)[C_{rj}(1,1)]$				$T_{rj}(1,2)[C_{rj}(1,2)]$			
r	f		r	f			
	1	2		1	2	3	
1	10 [5]	—	1	10 [5]	—	—	
4	—	50 [10]	2	—	20 [10]	—	
			3	—	—	10 [5]	

Product 2			
$T_{rj}(2,1)[C_{rj}(2,1)]$			
r	f		
	1	2	3
2	—	10 [10]	—
3	10 [15]	—	—
4	20 [20]	—	20 [10]

$h_{it}(1) = h_{it}(2)$			
t	t'		
	1	2	
1	0	5	
2	20	0	

Table 6. Process plans obtained in example (Case 2).

k [process plan]	p	d	t	t'	$X(kpdtt')$
1[(1,1),(2,4)]	1	1	1	1	4
1[(1,1),(2,2),(3,3)]	1	2	2	1	6
1[(1,1),(2,4)]	1	1	2	2	6
1[(1,1),(2,2),(3,3)]	1	2	2	2	14
2[(1,3),(2,2),(3,4)]	2	1	1	1	20
2[(1,3),(2,2),(3,4)]	2	1	2	2	10

Table 7. Workload distributions in machines in example (Case 2).

Resource	Planning period, t	
	1	2
1	100	200
2	320	380
3	260	240
4	600	500

ing an alternative design or process plan may increase the processing costs.

This also shows that a process plan or design with a higher processing cost can be selected by the model if it yields a lower overall cost (sum of the late shipment cost, inventory holding cost and processing cost).

Table 8. Problem sets tested.

Problem	P	R	T	Basis size	F_{pd}	D_p	R_{pd}
1	10	10	5	100	5	5	5
2	10	20	4	120	5	5	5
3	20	10	4	120	5	5	5
4	20	20	3	120	5	5	5
5	30	10	3	120	5	5	5
6	10	10	5	100	10	5	5
7	10	20	4	120	10	5	5
8	20	10	4	120	10	5	5
9	20	20	3	120	10	5	5
10	30	10	3	120	10	5	5

5. Computational experience

In this section computation experience of the column generation scheme is reported for large problems. The revised simplex algorithm incorporating the proposed column generation scheme was coded in Fortran and executed on a Sun Sparc station. Data for the test problems were generated as follows (Table 8).

Number of product types (P), time periods (T) and resources (R). Different combinations of these variables are taken (Table 8). The resulting basis size [given by $(P + R) \times T$], number of designs (D_p) and total number of resources in time (cost) matrix, (R_{pd}) are also shown.

Number of design features (F_{pd}). These are kept at two levels, 5 and 10.

Available time on resources (B_{rt}). This is kept as 10 000 time units for each resource at each time period.

Product demand (D_{pt}). The values for demand are generated from a uniform distribution between 25 and 50. However, the demand actually may not have uniform distribution. The same applies for other data generated.

Cost [$C_{rj}(pd)$] and time [$T_{rj}(pd)$]. These are generated from uniform distributions between 5 and 20.

Late shipment and inventory costs [$h_{it}(p)$]. These are generated from a uniform distribution between 5 and 10.

A total of 10 problems are generated (Table 8) and solved. Table 9 reports the computational experience of these problems.

The number of explicit columns, the number of times the generated columns entered the basis, the number of times problem SP is solved, the number of simplex iterations and the CPU time are reported. The solution time appears to increase greatly with increase in the number of columns. All the problems were solved within 35 min of CPU time.

6. Discussion

Model P0, developed in Section 2, considers minimization of the sum of the processing cost, late shipment cost and inventory cost. This model illustrates the importance of considering integration of design, process plan and production planning activities. Some other objectives may also be relevant in certain situations. For instance, there may exist differences in the selling price of two variations (designs) of the same product. To model this we can define $s(pd)$ as the unit selling price of product type p produced in design d and formulate model P1.

P1:

$$\begin{aligned}
 \text{Maximize } z_1 = & \sum_r \sum_{r'} \sum_p \sum_d \sum_k \sum_j \sum_{t'} s(pd) X(kpdtt') \\
 & - \sum_r \sum_{r'} \sum_p \sum_d \sum_k \sum_j \sum_{t'} A_{rj}(kpdtt') \\
 & \times C_{rj}(pd) X(kpdtt') - \sum_r \sum_{r'} \sum_p \sum_d \\
 & \sum_k h_{it}(p) X(kpdtt') \tag{6}
 \end{aligned}$$

Subject to: constraints (2)–(4).

Table 9. Computational results

Problem	Number of explicit x columns	Number of times x columns entered	Number of times SP solved	Total number of simplex iterations	CPU time (seconds)
1	3.9×10^6	50	12 750	50	36.0
2	2.5×10^6	40	8 200	40	49.6
3	5.0×10^6	81	32 800	81	102.9
4	2.8×10^6	62	18 900	62	77.6
5	4.2×10^6	278	127 350	282	351.8
6	1.2×10^{10}	140	35 750	142	110.7
7	0.8×10^{10}	43	8 800	43	53.6
8	1.6×10^{10}	901	360 400	901	1215.8
9	0.9×10^{10}	552	166 500	554	729.0
10	1.3×10^{10}	1536	691 650	1536	2112.9

The structures of model P0 and P1 are similar, and the column generation scheme developed to solve P0 can be suitably adapted for solving P1.

7. Conclusions

Product design, process planning and production planning are often considered independently. However, in order to effectively respond to the changes in business situations, such as changes in demand forecast, product mix and technology, it becomes desirable to consider them concurrently.

We have developed a mathematical model which concurrently selects product design, generates process plans and develops production plans to minimize the sum of processing cost, late shipment cost and inventory holding cost. It has been shown that the number of process plans can be large and therefore an efficient column generation scheme has been developed. The model and solution procedure are illustrated with examples. A problem with over 1×10^{10} variables has been solved on a Sun Sparc station within 35 min. A possible variation of the model is also presented that considers the difference in selling price which may exist for products made in different designs.

Acknowledgement

The author would like to thank the anonymous referers for their suggestions.

References

- ARCELUS, F. J., and SRINIVASAN, G., 1995, On the integration of production, inventory and trade-credit policies, *Production Planning & Control*, **6**, 455–460.
- DEWHURST, P., and BOOTHROYD, G., 1988, Early cost estimation in product design, *Journal of Manufacturing Systems*, **7**, 183–191.
- ELMARAGHY, H. A., and ELMARAGHY, W. H., 1992, Bridging the gap between process planning, production planning and control, in *Proceedings of 24th CIRP International Seminar on Manufacturing Systems*, Copenhagen, Denmark, 11–12 June, pp. 1–10.
- HORVATH, M., MARKUS, A., and VANCZA, J., 1996, Process planning with genetic algorithms on results of knowledge-based reasoning, *International Journal of Computer Integrated Manufacturing*, **9**, 145–166.
- HUANG, S. H., ZHANG, H. C., and SMITH, M. L., 1995, A progressive approach for the integration of process planning and scheduling, *IIE Transactions*, **27**, 456–464.
- LARSEN, N. E., and CLAUSEN, J., 1992, Applied methods for integration of process planning and production planning and production control, *Manufacturing International*, 349–364.

RAJAMANI, D., SINGH, N., and ANEJA, Y. P., 1996, Design of cellular manufacturing systems, *International Journal of Production Research*, **34**, 1917–1928.

Appendix. Illustration of column generation scheme

Consider the following problem.

$$P = 1; D_1 = 2; F_{11} = F_{12} = 2; T = 2; B_{it} = 1000 \quad \forall (r, t); \\ D_{11} = D_{12} = 30.$$

Processing cost and time matrices for the two designs are as follows.

Design 1

$$T_{rf}(1,1)[C_{rf}(1,1)]$$

r	f	
	1	2
1	20 [5]	9999 [9999]
2	9999 [9999]	50 [10]

Design 2

$$T_{rf}(1,2)[C_{rf}(1,2)]$$

r	f	
	1	2
1	20 [5]	9999 [9999]
2	9999 [9999]	30 [15]

Inventory or late shipment cost:

$$h_{it} = \begin{bmatrix} 0 & 5 \\ 20 & 0 \end{bmatrix}$$

Iteration 0: The size of the basis is 8. After introducing excess variables (e), artificial variables (a) and slacks (s) the constraints are:

$$\sum_r \sum_d \sum_k X(k, 1, d, t, 1) + a_1 - e_1 = 30$$

$$\sum_r \sum_d \sum_k X(k, 1, d, t, 2) + a_2 - e_2 = 30$$

$$\sum_r \sum_d \sum_k \sum_f A_{1,f}(k, 1, d, 1, t') T_{1,f}(1, d) \\ \times X(k, 1, d, 1, t') + s_3 = 1000$$

$$\sum_r \sum_d \sum_k \sum_f A_{2,f}(k, 1, d, 1, t') T_{2,f}(1, d) \\ \times X(k, 1, d, 1, t') + s_4 = 1000$$

$$\sum_r \sum_d \sum_k \sum_f A_{3,f}(k, 1, d, 1, t') T_{3,f}(1, d) \\ \times X(k, 1, d, 1, t') + s_5 = 1000$$

$$\sum_{r'} \sum_a \sum_k \sum_j A_{1,f}(k, 1, d, 2, t') T_{1,f}(1, d) \times X(k, 1, d, 2, t') + s_6 = 1000$$

$$\sum_{r'} \sum_a \sum_k \sum_j A_{2,f}(k, 1, d, 2, t') T_{2,f}(1, d) \times X(k, 1, d, 2, t') + s_7 = 1000$$

$$\sum_{r'} \sum_a \sum_k \sum_j A_{3,f}(k, 1, d, 2, t') T_{3,f}(1, d) \times X(k, 1, d, 2, t') + s_8 = 1000$$

Basic variables (BV) = $\{a_1, a_2, s_3, s_4, s_5, s_6, s_7, s_8\}$
 Cost vector (CV) = $\{M, M, 0, 0, 0, 0, 0, 0\}$ (where $M = 90\,000$ was taken).
 Basis (B) = identity matrix [I]
 Simplex multiplier = $[CV][B^{-1}] = [CV][I] = [CV]$.
 Or,
 $\{U_{11} = U_{12} = M, V_{11} = V_{21} = V_{31} = V_{12} = V_{22} = V_{32} = 0\}$

Iteration 1. There are a total of $4 \times 1 \times 2 \times 2 \times 2 = 32$ explicit columns. The number of process plans (the range of k) = $2^2 = 4$ for each design. We will not enumerate columns over k but will find the one for the most promising k^* by solving problem SP. For instance, consider design $d = 1$.

Solving SP is equivalent to computing the following matrix and then selecting the minimum in each column.

$$\begin{bmatrix} 5 - 20 \times 0 & 9999 - 9999 \times 0 \\ 9999 - 9999 \times 0 & 10 - 50 \times 0 \end{bmatrix} = \begin{bmatrix} 5^* & 9999 \\ 9999 & 10^* \end{bmatrix}$$

In other words, for each feature (column) find the resource (corresponding to the minimum row entry marked by $*$) to define the process plan k^* . This gives, $U(1, 1) = 5 + 10 = 15$ for $k^* = 1$ (number it as 1, say).

By following a similar process, we obtain $U(1, 2) = 20$. Thus design $d^* = 1$ is selected. The most promising process plan 1 for the selected design 1 can be described by the following binary matrix:

$A_{r,f}(1, 1, 1, t')$ matrix

r	f	
	1	2
1	1	0
2	0	1

The process plan can also be described as $1[(1, 1), (2, 2)]$.

Now we compute t^* and t'^* .

$F(1) = 15 + \text{minimum} [\text{minimum}\{(0-M), (5-M)\}, \text{minimum}\{(20-M), (0-M)\}]$ or, $F(1) = 15 - M$; (the two combinations $t = 1, t' = 1$ and $t = 2, t' = 2$ have a tie).

Let us pick the first combination, i.e. $t^* = 1$ and $t'^* = 1$.

The entering variable is $X(1, 1, 1, 1, 1)$ and entering column = $[1, 0, 20, 0, 50, 0, 0, 0]$.

Iteration 2.

BV = $\{a_1, a_2, s_3, s_4, X(1, 1, 1, 1, 1), s_6, s_7, s_8\}$
 CV = $\{M, M, 0, 0, 15, 0, 0, 0\}$
 $\{U_{11} = U_{12} = M, V_{11} = V_{21} = 0, V_{31} < 0, V_{12} = 0, V_{22} = V_{32} = 0\}$
 Variable $X(1, 1, 1, 2, 2)$ enters with $F(1) = 15 - M$.
 The entering column is $[0, 1, 0, 20, 0, 50, 0, 0]$.

Iteration 3.

BV = $\{a_1, a_2, s_3, s_4, X(1, 1, 1, 1, 1), X(1, 1, 1, 2, 2), s_7, s_8\}$
 CV = $\{M, M, 0, 0, 15, 15, 0, 0\}$
 $\{U_{11} = U_{12} = M, V_{11} < 0, V_{21} < 0, V_{31} = V_{12} = -M, V_{22} = V_{32} = 0\}$
 Variable $X(2, 1, 2, 1, 1)$ enters with $F(1) = 20 - M$.
 Selected process plan is $2[(1, 1)(2, 3)]$. The entering column is $[1, 0, 20, 0, 0, 0, 30, 0]$.

Iteration 4.

BV = $\{X(2, 1, 2, 1, 1), a_2, s_3, s_4, X(1, 1, 1, 1, 1), X(1, 1, 1, 2, 2), s_7, s_8\}$
 CV = $\{20, M, 0, 0, 15, 15, 0, 0\}$
 $\{U_{11} = 20, U_{12} = M, V_{11} = V_{21} = 0, V_{31} < 0, V_{12} < 0, V_{22} = V_{23} = 0\}$
 Variable $X(2, 1, 2, 2, 2)$ enters with $F(1) = 20 - M$.
 The entering column is $[0, 1, 0, 20, 0, 0, 0, 30]$.

Iteration 5.

BV = $\{X(2, 1, 2, 1, 1), X(2, 1, 2, 2, 2), s_3, s_4, X(1, 1, 1, 1, 1), X(1, 1, 1, 2, 2), s_7, s_8\}$
 CV = $\{20, 20, 0, 0, 15, 15, 0, 0\}$
 $\{U_{11} = 20, U_{12} = 20, V_{11} = V_{21} = 0, V_{31} < 0, V_{12} < 0, V_{22} = V_{32} = 0\}$

At this iteration it was found that no X -column could enter as $F(1)$ was equal to 0. Also, no slack or excess variable can enter so the procedure terminates. The optimal solution was

k [process plan]	p	d	t	t'	$X(kpdt')$	Processing cost	Inventory/late shipment cost	Plan cost
$1[(1, 1), (2, 2)]$	1	1	1	1	20	300	0	300
$1[(1, 1), (2, 2)]$	1	1	2	2	20	300	0	300
$2[(1, 1), (2, 3)]$	1	2	1	1	10	200	0	200
$2[(1, 1), (2, 3)]$	1	2	2	2	10	200	0	200
Total cost								1000