An Embedding Formulation For The Optimal Scheduling And Design Of Multi-Purpose Batch Plants

by

J. A. Vaselenak, I. E. Grossmann

EDRC-06-04-86 2

September 1986

AN EMBEDDING FORMULATION FOR THE OPTIMAL SCHEDULING AND DESIGN OF MULTI-PURPOSE BATCH PLANTS

Jane A. Vaselenak,[+] Ignacio E. Grossmann[*]
and Arthur W. Westerberg

Department of Chemical Engineering
Carnegie-Mellon University
Pittsburgh, PA 15213

September 1985/Rev. May 1986

[*]Author to whom correspondence should be addressed.

[+]Current Address: Shell Development Co., Houston, TX.

Abstract

This paper presents a new formulation for the optimal design of multipurpose batch plants in which not all products require the same processing equipment. In order to circumvent the combinatorial problem of selecting the product grouping for the optimal schedule, a superstructure representation is proposed that can be modeled as a multiperiod optimization problem. For most cases this problem can be condensed into a merged formulation that is similar to the problem of optimal design of multiproduct batch plants. The proposed method is very efficient as it requires the solution of a single MINLP problem. Two numerical examples are presented.

Introduction

This paper presents a novel method for the optimal design of multi-purpose batch facilities in which the types of equipment used for each product are specified. Products made in this type of manufacturing facility do not use the same equipment in the plant, and thus, can be separated into partitions for parallel processing with the restriction that all items of each batch stage are devoted only to one product at a time. A schedule or manufacturing order of the products is known as a "product configuration" (see Suhami and Mah, 1982). Because the scheduling of the products is a central feature of the design of this type of plant, this design problem is not a straightforward extension of the optimal design problem of multi-product batch plants (see Sparrow *et al.* 1975; Grossmann and Sargent, 1979). The number of possible product configurations for a given set of products can be rather large, leading to a difficult combinatorial problem for the optimal design.

Suhami and Mah (1982) have proposed a method where a lower bound is determined first for the minimum investment cost of multipurpose plants. Their strategy then applies rules to screen several random product configurations (typically ten candidates). Next for each of these alternatives a mixed-integer nonlinear program is solved by incorporating horizon constraints of the corresponding alternatives. The lowest total capital cost solution is retained, provided its value is within 10% of the lower bound estimate. Klossner and Rippin (1984) have also addressed this design problem. Their solution procedure is to examine all combinations of product groupings (i.e. all product configurations), by solving a mixed-integer nonlinear program to obtain the lowest total cost plant design for each configuration, and retain the best solution. The main disadvantage of these two procedures is that many different alternatives may have to be examined, each of which requires the solution of a mixed-integer nonlinear program.

The algorithm proposed in the present work overcomes this drawback by using a super-structure that embeds all of the groupings of products that are candidates for the optimal schedule, thereby requiring a single mixed-integer nonlinear program to be solved to obtain the plant design with the minimum total cost. With the super-structure, a multi-period model is formulated and then condensed into a merged formulation with equivalent horizon constraints. Interestingly, this merged formulation is similar in size to the problem of optimal design of multi-product batch plants. When equivalent horizon constraints cannot be found, which is an unusual case, a formulation that partially reduces the multi-period model can be derived. The time involved in identifying the super-structure and in writing the merged or partially merged formulation is minimal. Thus, the time savings in solving just one mixed-integer nonlinear program makes the proposed procedure computationally very efficient. Also, the global optimum solution is guaranteed for the relaxed merged formulation since the corresponding nonlinear programming formulation can be transformed to a geometric programming problem which has a unique optimizer.

Example of Grouping of Products

In this section a small example is considered to demonstrate how a super-structure can be developed that contains all of the potential alternatives for grouping of products. Figure 1 shows a plant consisting of four stages that is used to manufacture products A, B, C, and D. Product A uses stages 1 and 2; B, stages 2 and 3; C, stage 3; and D, stages 1 and 4. Products that do not share equipment stages may be processed in parallel. In Figure 2, the eight ways to schedule these products are presented. In product configuration 1 each product is manufactured individually. This option utilizes the equipment inefficiently when compared to the case of parallel processing because stage 3 is idle while product A is being processed (product C can be made at the same time), and stage 4 is empty while product B is being manufactured. Therefore, the sequential production of all four products can be excluded because (1) a longer production time would result for a fixed equipment or (2) larger equipment is required for a fixed total production time. However, product configurations 2, through 8, that employ the simultaneous manufacture of two products, are candidate schedules for this example. Product configuration 2 has product C processed at the same time as product D; configuration 3 has product B with product D; configuration 4 has product D first with product B and then with product C, etc. Since configuration 8 has embedded configurations 2 through 7, it represents a "super-structure" for these scheduling alternatives. Although configuration 3 is suboptimal since products A and C can be made at the same time, and although configuration 5 is also suboptimal because the manufacture of products B and D can be performed simultaneously, these configurations are also contained in the super-structure of Figure 2.

The superstructure representation of Figure 2 can be viewed as a multi-period model, with each period being devoted to a different group of products as shown in Figure 3. In particular, products A and C may be made at the same time as shown by the two arrows in period 1; the potential simultaneous manufacture of products B and D is portrayed by the two arrows contained in period 2; the manufacture of products C and D simultaneously is depicted by the two arrows of period 3. The length of each period, T^ T_2, T_3, is a variable and represents the maximum time for the simultaneous manufacture of the products in the group. If the time for a period is zero, then the group of products specified by that period should not be manufactured in parallel. For example, the results of permitting T_3, the length of period 3 to be zero, would correspond to configuration 7 in Figure 2.

Also, the length of the arrow for a product in the super-structure of Figure 3 represents the time needed for the manufacture of that product in the particular time period. In the example of Figure 3, T!, *Tl.* T?$_f$ T*, T^, and T^ are the variables relating the times needed for manufacture of the products in the three periods, in this example, if product configuration 2 in Figure 2 were the optimal schedule, then T^ and T^*_D would be zero; T^, T^*_B, TJ? and T^3 would have non-zero values.

Super-structure Determination

In order to generalize the derivation of super-structures for candidate product groups, it will be assumed that the multi-purpose batch plant consists of M pre-specified types of batch equipment R_j used for the processing of N different products. The manufacture of each product P_i utilizes $k_i \leq$ M types of batch equipment with each type corresponding to a stage R_j = 1, 2, ..., M. For each product, C_i is defined as the set of equipment types R_j used in the processing of product P_i.

$$C_i = \{ R_j \mid R_j \text{ required for the manufacture of product } P_i \} \quad (1)$$

$$|C_i| = k_i \qquad i = 1, 2, \ldots, N$$

The goal of the super-structure is to embed all the product groups that can be produced simultaneously. Each of these groups will be represented by a time period, as discussed in the example of the previous section. The following steps outline the basis of the method:

1. All binary combinations of products P_i and P_j are compared. If the binary combination does not share the same equipment types, the products P_i and P_j can be placed in the same group.

$$L_t = \{P_i, P_j\} \quad (2)$$

where L_t = a set of binary compatible products with product i, t=1,2,...

2. For each product P_i its corresponding groups L_t are checked for compatible production with a third product. If this condition is satisfied, a three product group is generated.

3. Each group of the three members is examined to see if a fourth member can be added, etc. This process stops when no more members can be added to any of the groups.

4. Groups that are subsets of another group are deleted. The remaining groups L_t are numbered and become the basis of the super-structure for the multi-period model.

It should be noted that the groups L_t, t = 1,2,...,T, where T is the number of periods, are identical to the subset of maximal sets I_j used by Imai and Nishida (1984) in their set partitioning formulation (Garfinkel and Nemhauser, 1972) for generating a near optimal configuration based on the heuristic of Suhami and Mah (1982).

Example of **Determining** the **Product Group for** the Super-Structure

The procedure described in the previous section for deriving superstructures will be illustrated with the example problem of Suhami and Mah (1982). The product-equipment incidence matrix, denoted by A, is given in Table I.

Step 1. Check all binary combinations of products for compatible production. The result is shown below:

$$
\begin{aligned}
L_1 &= \{P_A, P_B\} \\
L_2 &= \{P_A, P_D\} \\
L_3 &= \{P_A, P_E\} \\
L_4 &= \{P_A, P_F\} \\
L_5 &= \{P_B, P_G\} \\
L_6 &= \{P_C, P_D\} \\
L_7 &= \{P_E, P_F\} \\
L_8 &= \{P_F, P_G\}
\end{aligned}
\tag{3}
$$

Step 2. Test the above sets for additions of another compatible product. From L_3, L_4, and L_7, the only three-product set obtained is,

$$L_9 = \{P_A, P_E, P_F\} \tag{4}$$

Step 3. Examine the remaining products for parallel processing with the resulting set of step 2. Since no more products can be added, the enumeration phase is completed.

Step 4. Delete the sets that are subsets of other listed sets. In this example, $\{P_A, P_E\}$, $\{P_A, P_F\}$ and $\{P_E, P_F\}$ are deleted because they are subsets of $\{P_A, P_E, P_F\}$

The sets used for the super-structure are listed below, and shown in Figure 4 in the multiperiod representation:

$$
\begin{aligned}
L_1 &= \{P_A, P_B\} \\
L_2 &= \{P_A, P_D\} \\
L_3 &= \{P_A, P_E, P_F\} \\
L_4 &= \{P_B, P_G\} \\
L_5 &= \{P_C, P_D\} \\
L_6 &= \{P_F, P_G\}
\end{aligned}
\tag{5}
$$

MINLP Formulation of Multi-period Model

Having addressed the problem of finding a suitable representation for the alternative schedules or product configurations, the problem of determining the optimal number of units and their capacities that minimize the total capital cost of the batch equipment will be considered. This plant must be able to accommodate the desired production goals of all the products within a specified time. The cost of the semi-continuous equipment, i.e., pumps, heat exchangers, etc., is assumed to be negligible compared to the cost of the batch vessels or to be relatively constant, and therefore, is excluded from this analysis. For scheduling the products for parallel processing, the super-structure from the previous section is used.

The following mixed-integer nonlinear program can be formulated for the superstructure. The mufti-purpose batch plant is used to process N different products using M pre-specified types of batch equipment R_j. The objective for the optimal design is then to minimize the total capital cost of the batch equipment:

$$
\min \sum_{j=1}^{M} \alpha_j N_j (V_j)^{\beta_j} \tag{6}
$$

where N_j = the number of units of equipment type R_j

V_j = the total volume of the unit of equipment type R_j

α_j, β_j = cost coefficients for equipment type R_j.

The quantity of product P_i that is made in each time period t, symbolized by q_i^t, is the product of the number of batches of product P_i in period t and its batch size B_i:

$$q_i^t = n_i^t B_i \qquad i = 1, 2, \ldots, N; \qquad t \in \ell_i \tag{7}$$

$$\ell_i = \{t \mid P_i \in L_t\} \tag{8}$$

The total amount of product P_i that is manufactured must meet or exceed its production goal, Q_i:

$$\sum_{t \in \ell_i} q_i^t = \sum_{t \in \ell_i} n_i^t B_i \geq Q_i \qquad i = 1, 2, \ldots, N \tag{9}$$

The required size of the equipment in the plant is assumed to be a function of the batch size of each product. In particular, the capacity needed for one batch of product P_i in a vessel of type R_j represented by v_{ij} is equal to the product of its size factor S_{ij} and its nominal batch size B_i:

$$v_{ij} = S_{ij} B_i \qquad i = 1, 2, \ldots, N; \qquad R_j \in C_i \tag{10}$$

$$B_i \geq 0 \qquad i = 1, 2, \ldots, N \tag{11}$$

The volume for any stage R_j must accommodate the maximum processing volume v_{ij} for all the products using stage j, denoted by the set U_j.

$$V_j = \max\{v_{ij}\} \qquad j = 1, 2, \ldots, M; \qquad i \in U_j \tag{12}$$

$$U_j = \{P_i \mid R_j \in C_i\} \tag{13}$$

This constraint can be written as follows:

$$V_j \geq v_{ij} \qquad j = 1, 2, \ldots, M; \qquad i \in U_j \tag{14}$$

where the volume must be selected within the specified range.

$$V_j^L \leq V_j \leq V_j^U \qquad j = 1, 2, \ldots, M \tag{15}$$

where V_j^L = lower limit for equipment of type R_j

V_j^U = upper limit for equipment of type R_j

Since the same batch size B_i is used throughout the manufacture of product P_i, all the items of a stage j have the same capacity V_j. Each stage j consists of N_j units operating independently in parallel and out of phase, thus affecting the cycle time of the stage. N_j is restricted to integer values and limited by,

$$N_j^L \leq N_j \leq N_j^U \qquad j = 1, 2, \ldots, M \tag{16}$$

where N_j^L = lower limit on equipment items for stage j

N_j^U = upper limit on equipment items for stage j

T_{Li}, the time required for the manufacture of a batch of product P_i, is the maximum stage cycle time of the stages used for making product P_i (see Grossmann and Sargent, 1979):

$$T_{Li} = \max \{ (t_{ij}) / (N_j) \} \qquad i = 1, 2, \ldots, N; \; R_j \in C_i \tag{17}$$

This constraint can be replaced by the following inequalities:

$$T_{Li} \geq [(t_{ij}) / (N_j)] \qquad i = 1, 2, \ldots, N; \; R_j \in C_i \tag{18}$$

Since T_{Li} depends only on the number of units in each stage used by product P_i, and since the equipment in the plant is fixed during the horizon or the total time available for processing being considered, T_{Li} does not vary with the time periods t.

The time required to manufacture a product P_i in period t is given by the number of batches in period t, n_i^t, multiplied by the time needed per batch, T_{Li}. Therefore, T_t, the time of a period t, must be greater than or equal to the time needed in manufacturing the products in period t:

$$T_t \geq n_i^t \, T_{Li} \qquad t = 1, 2, \ldots, T; \qquad P_i \in L_t \tag{19}$$

Finally, the sum of the lengths of the periods must be less than or equal to the total available production time H.

$$\sum_{t=1}^{T} T_t \leq H \tag{20}$$

The problem described by Equations (6)-(11), (14)-(16), and (18)-(20) corresponds to a MINLP formulation for the proposed multi-period model. The relationship between the super-structure to the mixed-integer nonlinear program presented above is apparent. However, because of its large size, this model cannot be solved easily. In addition to the fact that the number of variables and the number or constraints in the problem can be large, the existence of a unique optimizer is not necessarily guaranteed.

The presence of multiple minimizers can be seen easily. From the super-structure of Figure 3, product C can be processed with products A or D or with both products. As shown in Figure 5, assume that the optimal production time of product D is greater than that of A or of B, which in turn are greater than that of C. Because the manufacture of A, B, and D dominates the plant, the processing of product C can occur in period 1 or in period 3 or can be split between the two periods. These options are shown in Figure 5. In the first case T_C^1 has a non-zero value and T_C^2 is zero. For the second alternative, T^ is zero and T^ is non-zero. In the third alternative, both T_r^1 and T!? are non-zero and can have an infinite number of values. However, note that the design of the plant is unaffected by changes -in these variables, and therefore, many different solutions yield the same total cost. Although this problem has non-unique solutions, a merged formulation can be found that circumvents this difficulty as described in the following section.

Merged Formulation for a Multi-Purpose **Batch Plant**

By analyzing the different cases in Figure 5, it is apparent that the total time required to manufacture each product has a unique value. Therefore, this fact suggests that a "merged" formulation can be developed in which the total processing times and total number of batches for each product will be used as variables, instead of defining variables for each time period as was done in the previous formulation. This new merged formulation can be attained by reformulating constraints in the mixed-integer nonlinear program of the prior section.

The sum of the number of batches of each product in each period, n_i^l, can be replaced with the total number of batches of each product, n_i.

$$n_i = \sum_{t \in \ell_i} n_i^t \qquad i = 1, 2 \ldots N \tag{21}$$

By substituting n_i in the production demand constraints. Equation (9) then becomes

$$n_i B_i \geq Q_i \qquad i = 1, 2 \ldots N \tag{22}$$

For the merged formulation it is convenient to define T_i, the total processing time of product P_i, which is greater or equal to the total number of batches of P_i multiplied by the time per batch of P_i. The following constraint relates T_i with the limiting cycle time T_{Li}:

$$T_i \geq n_i T_{Li} \qquad i = 1, 2, \ldots, N \tag{23}$$

This constraint will be used in place of constraint (19) since the variables T_t will be excluded in the merged formulation. Finally, for this formulation the horizon constraint in Equation (20) must be replaced by equivalent horizon constraints that are expressed in terms of the new variables T_i. To gain insight into how these constraints can be derived, consider the following example.

Example of Equivalent Horizon Constraints

For Example 1, whose super-structure is shown in Figure 3, the constraints from the multi-period model that define the lengths of the periods are:

$$T_1 \geq n_A^1 T_{LA} \tag{24}$$

$$T_2 \geq n_B^2 T_{LB} \tag{25}$$

$$T_1 \geq n_C^1 T_{LC} \tag{26}$$

$$T_3 \geq n_C^3 T_{LC} \tag{27}$$

$$T_2 \geq n_D^2 T_{LD} \tag{28}$$

$$T_3 \geq n_D^3 T_{LD} \tag{29}$$

The total time devoted to making each product is:

$$T_1 \geq n_A^1 T_{LA} = T_A \tag{30}$$

$$T_2 \geq n_B^2 T_{LB} = T_B \tag{31}$$

$$T_1 + T_3 \geq (n_C^1 + n_C^3) T_{LC} = T_C \tag{32}$$

$$T_2 + T_3 \geq (n_D^2 + n_D^3) T_{LD} = T_D \tag{33}$$

Since **the horizon** constraint in the multi-period **model** is:

$$T_1 + T_2 + T_3 \leq H \tag{34}$$

it is apparent that the following horizon constraints are equivalent to those of the multi-period formulation (see Figure 3):

$$T_A + T_B * H \quad \text{Periods: } 1,\ 2 \tag{35}$$

$$T_A * T_D * H \quad \text{Periods: } 1,\ 2,\ 3 \tag{36}$$

$$T_B + T_C * H \quad \text{Periods: } 1,\ 2,\ 3 \tag{37}$$

Inequality (35) is necessary for the case in which the production times for products A and B are are much longer than those for C or D at the optimum. Here, T_A determines the length of period 1 and T_B, period 2. Because products C and D can be manufactured at the same time as A and B, respectively, period 3 vanishes. Thus, constraint (35) is active and constraints (36) and (37) are inactive in this situation. On the other hand (36) and (37) are active constraints if A and D, or B and C, are dominant products, respectively.

The Appendix presents the general steps involved in the derivation of the horizon constraints for the merged formulation. As will be discussed later, it is not always possible to derive these constraints, in which case a partially merged formulation is needed.

Merged Formulation

Provided equivalent horizon constraints can be derived as those shown in the previous section, the multi-period model can be condensed into a merged formulation that is given by:

$$\min \sum_{j=1}^{M} *\backslash N_j (V_j \wedge j \qquad (38>$$

where N_j = the number of units of equipment type fl_j

V_j = the total volume of the unit of equipment type R_j

a_j, fi_j = cost coefficients for equipment type R_j.

subject to:

$$n_i B_i \geq Q_i \qquad i = 1, 2 \ldots N \tag{39}$$

$$V_J \geq S_{ij} B_i \qquad j = 1, 2 \ldots M; \quad i \in U_J \tag{40}$$

$$V_j^L \leq V_j \leq V_j^U \qquad j = 1, 2 \ldots M \tag{41}$$

where V_j^L = lower limit for equipment of type R_j

V_j^u = **upper** limit for equipment of type R_j

$$^T Li \geq [(V'(N_j)] \qquad i = 1, 2 \ldots N; \quad R_j G C^\wedge \tag{42}$$

$$T_i \leq n_i T_u \qquad i = 1, 2 \ldots N \tag{43}$$

$$N^\wedge \leq N_j \leq N^\wedge \qquad j = 1, 2 \ldots M \tag{44}$$

where N_j^L = lower limit on equipment items for stage j

N_j^u = upper limit on equipment items for stage j

Finally, equivalent horizon constraints that can be expressed as linear inequalities m **terms of T_i must be included.**

$$f(T_r \ldots T_i \ldots T_N) * H$$

The continuous relaxation of this formulation can be reformulated as a geometric **program and thus, it has a unique optimal solution (see Grossmann and** Sargent, **1979). Furthermore, the importance of the above formulation is that it** is essentially **identical to the multi-product design problem, except for the horizon** constraints. **Therefore, the significance of this model is that it reduces the multi-purpose** design problem to a single MINLP which avoids the combinatorial problem of testing alternative product configurations as in the methods of Suhami and Mah (1982) and Klossner and Rippin (1984).

It should be noted, however, that total merging of the multi-period problem cannot always be achieved because equivalent horizon constraints may not be found, as will be shown in the next section. In this case, which is an unusual occurence, it is possible to develop a partially merged formulation which is still substantially smaller than the multi-period model.

Partially Merged Formulation for "Cycle" Type Problems

The following example illustrates the case in which only a partial merging of the multiperiod model can be achieved. In particular, consider the super-structure shown in the form of a matrix in Figure 6 that contains a loop indicated by the dashed lines. A loop is a path that starts at a cell or non-zero entry in the matrix and by using a set of alternating horizontal and vertical moves through other non-zero cells, returns to the initial cell. If all the periods of this super-structure are non-zero as shown in Figure 7, then the schedule could be implemented as displayed in Figure 8. Note that there is no possible arrangement of the periods to prevent an interruption in the manufacture of the five products. In fact, in previous work, such as the algorithm of Suhami and Mah (1982), this possible schedule is not considered. For the lack of a more descriptive term, the problem of interrupted productions in a schedule is called a "cycle" problem. As shown in Figure 8 processing begins with product A in the first period and "cycles" back to product A again in the last period. The following time constraints can be derived for the super-structure given in Figure 7.

$$T_A \leq T_1 + T_2 \qquad (45)$$

$$T_B \leq T_1 + T_3 \qquad (46)$$

$$T_C \leq T_3 + T_4 \qquad (47)$$

$$T_D \leq T_4 + T_5 \qquad (48)$$

$$T_E \leq T_2 + T_5 \qquad (49)$$

If the procedure of the Appendix is used, the horizon constraints for the merged formulation for the super-structure would be as follows:

$$T_A + T\_ \;£\; H \qquad \text{periods: } 1.\ 2.\ 3,\ 4 \qquad (50)$$

$$T_A \bullet T_Q \;£\; H \qquad \text{periods: } 1,\ 2.\ 4.\ 5 \qquad (51)$$

$$T_Q + T_n * H \qquad \text{periods: } 1,\ 3,\ 4.\ 5 \qquad (52)$$

$$T_D \bullet T_t \leq H \qquad \text{periods: } 1,\ 3,\ 2.\ 5 \qquad (53)$$

$$T_c + T_E \leq H \qquad \text{periods: } 3,\ 4,\ 2,\ 5 \qquad (54)$$

However, it can be noted that none of the constraints given by Equations (50) to (54) implicitly contains all five periods. To illustrate further this situation, assume all periods are active at the solution. Also, assume the production of each product will fill its allotted periods as shown in Figure 7, and that the length of each period $<T_t)$ is 100 and the horizon (H) is 400. From constraints (45) through (49), the variables for processing times (T_i) equal 200 and satisfy the horizon constraints, (50) to (54). However, although all of the constraints are satisfied, the physical situation is not accurate. From Figure 8, it can be seen that for H=400, the length of each period (T_t) should be 80 and not 100, if all periods are to be equal in length, and that the processing times for each product T_i should be 160 and not 200. Because of these contradictions, this model must be revised using a partial merging of constraints m the multiperiod model as described in the next section.

Partial Merging of Product A

In order to avoid the inconsistency in the horizon constraints, consider that the batches of product A in its two time periods, which are responsible for the cycle in the schedule of Figure 8, are not merged. This situation is equivalent to treating product A as two products A' and A" in its corresponding periods, yielding the super-structure shown in Figure 9. With this splitting of product A, constraint (45) can be replaced with the following two constraints:

$$T_A^1 \le T_1 \qquad (55)$$

$$T_A^2 \le T_2 \qquad (56)$$

The production goals for product A are then expressed by the following constraint:

$$\left(n_A^1 + n_A^2\right) B_A \ge Q_A \qquad (57)$$

The batch sizes of product A' and of product A" will be assumed to be equal since they are the **same** product. Since the cycle times of A' **and** A" **are** equal also, it **follows that:**

$$\left(n_A^1\right) T_{LA} \le T_A^1 \qquad (58)$$

$$\left(n_A^2\right) T_{LA} \le T_A^2 \qquad (59)$$

The other constraints for product B, C, D, and E are the same as in the merged **formulation with the exception of the horizon constraints. Equations (50) to (54)** are **replaced with the following:**

$$T_A^1 + T_C + T_E \le H \quad \text{periods: } 1,\ 3,\ 4,\ 2,\ 5 \qquad (60)$$

$$T_A^2 + T_B + T_D \le H \quad \text{periods: } 2,\ 1,\ 3,\ 4,\ 5 \qquad (61)$$

$$T_Q + T_c \le H \quad \text{periods: } 1,\ 3,\ 2,\ 5 \qquad (62)$$

which now include all five time periods, and **therefore are valid horizon** constraints.

It can then be seen that by merging the batches for all of the initial products B, C, D, and E, and splitting product A, a partially merged formulation can be obtained with equivalent horizon constraints similar to the original multi-period problem. Clearly, the question that then arises is how to identify systematically products that require splitting in these cycle problems.

Partially Merged Formulation

This section presents an algorithm to determine product splitting for problems in which the merged horizon constraints are not equivalent or complete. In this case, the variables for at least one product must remain unmerged; the values for the individual periods in which that product appears are used instead of the combined variables. For instance, if product A appears in periods one and two, its production in the two periods must be treated as distinct variables. Another view of this procedure is to split product A into two new products, A' and A", whose total productions are related. Additional constraints are written for these newly defined products and their horizon constraints are inspected. Since at least one constraint implicitly contains all the time periods, then the constraints are a true representation of the physical situation and the resulting mixed-integer nonlinear program can be solved. On the other hand, if the horizon constraints are not complete, another product must be split into its multi-period form. The procedure is repeated until the horizon constraints are complete. The steps of this procedure are then as follows:

1. Derive the totally merged formulation, including the horizon constraints as indicated in the Appendix.

2. Check the horizon constraints to see that at least one constraint implicitly contains all the time periods. If this condition is met, solve the merged formulation. If not, go to step 3.

3. Select a product that breaks the loop(s) present in the matrix of the superstructure. Choose the fewest number of products for this step. A product that appears in the fewest number of time periods will add the fewest number of additional variables and constraints to the formulation. A product with a large Q_i will probably have good numerical stability when split into its values for the time periods.

4. Obtain a new super-structure using the artificial products for the product(s) selected in step 3. Derive the new horizon constraints to verify implicit presence of all the time periods in at least one constraint. If this state does not exist, return to step 3 and alter the selection of the product(s). Otherwise, derive the necessary multi-period constraints for the "split" products and solve the resulting partially merged formulation.

Discussion

It should be noted that constraints of the type shown in Equation (57) cannot be transformed into those for a geometric programming problem by using the exponential transformation of variables. Therefore, since the partially merged formulation involves non-convex constraints, the existence of a unique optimizer cannot be proven for the relaxed problem. However, the existence of a unique minimum value for the objective function can be proved as shown in Vaselenak (1985).

It should be pointed out that the existence of a loop in the matrix of the superstructure does not always imply that the merged formulation using the original products will not be valid. For example, it can be shown that the matrix of the super-structure for the six product example in Figure 10 contains a loop. However, the horizon constraints listed below, which do not require splitting of products, implicitly cover all six time periods:

$$T_A \bullet T\text{-} \bullet T_C \;£\; H \qquad \text{periods: } 1, 6, 2, 3, 4, 5 \qquad (63)$$

$$T_D + T_U \bullet T_r \;*\; H \qquad \text{periods: } 1, 2, 3. 4. 5. 6 \qquad (64)$$

Therefore, the existence of a loop in the superstructure matrix is a necessary but not a sufficient condition for the use of artificial products in a partially merged formulation. It is also important to emphasize that the occurrence of "cycle" problems that require splitting of products is quite rare. In actual practice most of the problems will not exhibit this behavior and, therefore, can be solved with the merged formulation.

Summary of the Basic Steps

The basic steps in the suggested procedure for the optimal design of multi-purpose batch plants can be summarized as follows:

1. List the candidate product groups.

2. Obtain the super-structure that contains the candidate groups.

3. Derive the horizon constraints.

4. Examine the horizon constraints. If at least one horizon constraint implicitly contains all the time periods, complete the merged formulation and solve the MINLP given in Equations (38) to (44) with the corresponding horizon constraints. Otherwise, go to step 5.

5. Select a product(s) that breaks the loop(s) in the super-structure. Treat the chosen product(s) as two or more artificial products.

6. Revise the super-structure to include the additional products. Verify that the loop is broken. If this condition is met, go to step 7. Otherwise, return to step 5 and modify the products chosen for splitting.

7. Derive the multi-period form of the constraints for the product(s) that are split into its (their) values for the time periods in which it (they) appear. Formulate the merged representation of the model for the rest of the products, including the horizon constraints. At least one horizon constraint will contain implicitly all the time periods.

Example 2

The seven product, ten stage example problem of Suhami and Mah (1982) has been solved to illustrate the design of a large multi-purpose batch plant. From the super-structure of this problem (Figure 4), the horizon constraints shown in Table II are derived.

Using these constraints, the example has been solved with the merged formulation. Data for this problem appears in Table III. In addition, the following specifications have been used: $H = 6200$ hr, $a = 250$ SFr, $fi = 0.6$, $1 \le N_j \le 3$, and $250 \le V_j$ £ 10,000. As seen in Table IV, the MINLP formulation involves 48 variables and 67 constraints. MINOS/Augmented (Murtagh and Saunders, 1980) required 16.09 seconds of CPU-time on a DEC-20 computer to solve the relaxed nonlinear program.

The following constraints were used to force the N_j to be integers:

$$[(N_j^*)^u - N_j][N_j - (N_j^*)^L] = 0 \qquad \text{for non-integer } N_j \tag{65}$$

where $(N_j^*)^u$ = the smallest integer greater than N_j^*

$(N_j^*)^L$ = the largest integer less than N_j^*

N_j^* = the non-integer solution for N_j

This type of constraint, although multi-modal, provides an efficient way to locate a feasible integer solution that is close to the relaxed solution, which represents a lower bound for the optimal integer solution. (In these examples the integer solutions are within approximately 1% or less of the continuous solutions.) A branch and bound procedure to search all combinations of the integer N_j, that involves the solution of a sequence of nonlinear programs, is an expensive alternative to the use of this constraint. If the constraints of the type shown above fail to find a feasible solution, then the branch and bound method may be necessary to find an integer solution. However, because the integer solutions obtained with constraint (65) are often close to their lower bounds, the expense of using a branch and bound method is usually not justified.

The solution for this problem is given in Table IV. The total computer time that was required is 35 sec (DEC-20). Note that the optimal objective function value differs only slightly with the one obtained by Suhami and Mah (1982). From the integer solution obtained, the plant can be scheduled as shown in Figure 11. This schedule has been determined by examination of the active horizon constraints. Since products A, E, and F have the same processing times and since these three products appear in period 3, T_3 is set equal to their value of 2159.2 hr. Similarly, products B and G determine the length of period 4, and products C and D, period 5. Comparing the schedule given in Figure 11 with the super-structure in Figure 4, periods 1, 2, and 6 of the super-structure have been deleted.

This example shows that with the merged MINLP formulation significant time savings are possible due to the fact that the combinatorial problem of analyzing alternative product groupings has been avoided totally with this new formulation.

Example 3

To illustrate an application of the partially merged formulation, the five product, five stage example of Figure 7 has been solved. Data for this problem are listed in Table V, and the superstructure after the splitting of A is shown in Figure 9.

In this example, H = 6200, α = 250 SFr, β = 0.6, $1 \le N_j \le 3$, and $250 \le V_j \le 10{,}000$. The horizon constraints are given by Equations (63)-(65). As presented in Table VI, the relaxed form of the partially merged formulation contains 34 variables and 35 constraints. The solution required a total of 59 CPU-sec on a DEC-20 computer using MINOS/Augmented to solve both the relaxed nonlinear program and the one with integrality constraints. The solution is displayed in Table VI. The optimal schedule for the integer solution is given in Figure 12 where period 2 has been placed at the end. The processing time for the first part of product A (i.e. A') in period 1 has been used to determine the length of this period. Next, product B has been considered. Its remaining processing time that does not fit into period 1 has fixed the length of period 3. Similarly, the processing time of product C has been split between periods 3 and 4. The lengths of periods 5 and 2 are set with the part of D that could not be accomodated in period 4, and with product E and the rest of product A (i.e. A").

In this schedule, each product is produced simultaneously with one product first and then with a second product. For example, product A is processed with product B first in period 1, and then B with product C in period 3 (see Figure 12). Also, note that the production of A is interrupted because it takes place in periods 1 and 2. As was indicated previously, the methods of Suhami and Mah (1982) and Klossner and Rippin (1984) cannot identify this optimal schedule.

Conclusions

This paper has presented a new approach for the optimal design of multipurpose batch plants. It has been shown that by embedding the possible product groupings for the schedule, the combinatorial problem of analyzing each alternative can be eliminated. For most cases the problem can be merged into a single MINLP problem that is very similar in structure to the design problem for multiproduct plants. When this merging is not possible, the formulation still leads to an MINLP problem of reasonable size. The numerical results show that this new approach is computationally efficient.

Acknowledgments

The authors gratefully acknowledge financial support from the National Science Foundation under Grants CPE-8210971 and CPE-8351237.

References

1. Garfinkel, R.S. and Nemhauser, G.L., *Integer Programming;* John Wiley, New York, 1972.

2. Grossmann, I.E., and Sargent, R.W.H., *Ind. Eng. Chem. Proc. Des. Dev..* 1979, *18.* 343-348.

3. Imai, M., and Nishida, N., *Ind. Eng. Chem. Proc. Des. Dev..* 1984, *23.* 845-847.

4. Klossner, J., and Rippin, D.W.T., "Combinatorial Problems in the Design of Multiproduct Batch Plants - Extension to Multiplant and Partly Parallel Operations", A.I.Ch.E. Annual Meeting, San Francisco, November 1984.

5. Murtagh, B. A., and Saunders, M. A. *MI NOSIAUGMENT ED User's Manual.* Technical Report SOL 80-14, Stanford University Systems Optimization Laboratory, June 1980.

6. Sparrow, R.G., Forder, G.J. and Rippin, D.W.T., *Ind. Eng. Chem. Proc. Des. Dev..* 1975, *14.* 197.

7. Suhami, I. and Mah, R S.H., *Ind. Eng. Chem. Proc. Des. Dev..* 1982, *21.* 94-100.

8. Vaselenak, J.A., Ph.D. Dissertation, Carnegie-Mellon University, Pittsburgh, PA, 1985.

APPENDIX

Determination of Horizon Constraints for Merged Model

The purpose here is to find all the combinations of products that may limit the total production time. Another way of viewing these horizon constraints is to consider them as replacements for the horizon constraints in the multiperiod model that involves the sum of lengths of the time periods. All possible combinations of the new variables, the total processing times for each product, T_A, T_B, etc., must be taken into account when eliminating the time length variables, *Ty* T''_2, etc., from the problem. This procedure is a recursive tree search with nodes only attached to nodes of a higher index, as illustrated in Figure A1. The steps in the procedure are as follows:

1. Determine the super-structure for the multi-purpose plant design problem.
2. Delete any products that appear in the same periods as the first product from consideration for constraints involving this first product.
3. From the subset of remaining products consider all combinations of this subset that do not share a common period. Use these particular combinations with the first product to list its horizon constraints. If this subset of products is the empty set, consider the processing time of the first product to be less than the total available time as the only possible constraint in this step.
4. Omit the first product from analysis. Repeat steps 2 through 4 with each subsequent product until all products have been considered.
5. Delete redundant constraints.
6. Check the final set of horizon constraints to see that at least one constraint contains all the time periods. If this condition is met, the merged formulation can be used. If none of the new horizon constraints has all of the periods, a non-typical case, then a partial merging of the constraints can be used, as explained in section 2.7.

From the example in Figure 3 the resulting tree search is shown in Figure A2, with "x" representing the exclusion of the rest of the branch. In the branch ABC, products A and B do not have any time periods in common, but C is contained in period i with product A. Thus, the branch AB is a viable constraint.

$$T_A \bullet T_B \lhd H \quad \text{(A1)}$$

The other two horizon constraints are shown in branches AD and BC.

$$T_A \bullet ' T_D * H \quad \text{(A2)}$$

$$T_B \bullet T_c \pounds H \quad \text{(A3)}$$

The constraint from branch C, T'_c £ H, is deleted since it is redundant due to constraint (A3). Also, the constraint from branch D, T_D £ H, is redundant because of constraint (A2).

Table I: Matrix A for the Example

	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}
P_A	0	0	0	0	0	0	0	1	0	0
P_B	0	0	1	1	0	0	1	0	1	0
P_C	0	0	1	0	0	0	0	1	1	1
P_D	0	0	0	1	1	0	1	0	0	0
P_E	1	0	0	0	1	0	0	0	1	0
P_F	0	0	1	0	0	1	1	0	0	1
P_G	1	1	0	0	1	0	0	1	0	0

Table II: Horizon Constraints for Example 2, Suhami and Mah (1982)

$$T_A + T_C + T_G \leq H$$
$$T_B + T_C + T_E \leq H$$
$$T_B + T_C + T_F \leq H$$
$$T_B + T_D + T_E \leq H$$
$$T_B + T_D + T_F \leq H$$
$$T_C + T_E + T_G \leq H$$
$$T_D + T_E + T_G \leq H$$

Table III: Data for Example 2

a) Processing times, t_{ij} (hr/batch)

Product	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}
					Unit type					
P_A								7.456		
P_B			7.143	2.595			3.974		5.719	
P_C			4.36					2.554	7.318	2.297
P_D				2.404	9.987		6.758			
P_E	6.534				5.516				1.932	
P_F			1.269			2.005	5.469			7.725
P_G	6.855	7.326			5.062			6.65		

b) **Size** factors, S_{ij} (L/kg/batch)

Product	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}
					Unit type					
P_A								5.404		
P_B			9.768	1.125			3.205		3.304	
P_C			8.065					4.62	4.529	8.163
P_D				1.922	9.415		4.833			
P_E	9.422				2.653				5.982	
P_F			3.174			2.895	5.731			3.587
P_G	3.757	5.64			9.381			6.418		

c) **Production Goals (kg)**

Product	**Yearly production Q_i**
P_A	300,000
P_B	150,000
P_C	200,000
P_D	190,000
P_E	140,000
P_F	172,000
P_G	106,000

Table IV: Solution of Example 2

Stage	Continuous V_j (liters)	Continuous N_j	Integer V_j (liters)	Integer N_j
1	3789.0	1.000	3991.6	1.000
2	2976.2	1.000	3040.8	1.000
3	7112.0	1.000	7266.4	1.000
4	1331.1	1.000	1402.5	1.000
5	6521.3	1.116	6870.1	1.000
6	1691.0	1.000	1781.5	1.000
7	3347.6	1.000	3526.6	1.000
8	5313.9	1.000	5598.1	1.000
9	2405.6	1.123	2549.0	1.000
10	4335.9	1.000	4594.4	1.000

Integer Solution

Product i	n_i	B_i (kg)	T_{Li} (hr)	T_i (hr)
A	289.6	1036.	7.456	2159.2
B	201.6	743.9	7.143	1440.3
C	355.3	562.8	7.318	2600.4
D	260.4	729.7	9.987	2600.4
E	330.5	423.7	6.534	2159.2
F	279.5	615.4	7.725	2159.2
G	196.6	539.2	7.326	1440.3

	Continuous	Integer
Objective function value (this study)	354,778	355,516
Objective function value (Suhami and Mah, 1982)	354,770	355,505
Number of variables	48	48
Number of constraints	67	69
CPU-time (DEC-20 computer)	16.09	18.79
% deviation from continuous solution		0.2083 %

Table V: Data for Example 3

a) Processing times, t_{ij} (hr/batch)

		Stage				
		1	2	3	4	5
Product	A	9.0	6.0			
	B			3.9	6.2	
	C		5.5			3.5
	D	7.5		4.5		
	E				7.1	4.0

b) Size Factors, S_{ij} (L/fcg/batch)

		Stage				
		1	2	3	4	5
Product	A	3.0	2.5			
	B			1.0	1.5	
	C		2.7			2.3
	D	3.1		1.1		
	E				1.7	2.8

c) Production Goals (Kg/yr)

	Product				
	A	B	C	D	E
Production goal (Jcg/yr)	300,000	195,000	220,000	190,000	170,000

Table VI: Solution of Example 3

Stage	Continuous V_j (Liters)	Continuous N_j	Integer V_j (Liters)	Integer N_j
1	1817.7	1.1183	2029.0	1.0000
2	1514.8	1.0000	1690.8	1.0000
3	645.0	1.0000	720.0	1.0000
4	783.4	1.2075	928.2	1.0000
5	1290.4	1.0000	1440.3	1.0000

Integer Solution

Product i	n_i	B_i (kg)	T_{Li} (hr)	T_i (hr)
A'	213.5	676.3	9.000	1921.5
A"	230.1	676.3	9.000	2070.5
B	314.9	618.8	6.200	1952.3
C	351.3	626.2	5.500	1932.1
D	290.3	654.5	7.500	2177.1
E	330.5	514.4	7.100	2346.3

	Continuous	Integer
Objective function value	92,455	93,413
Number of variables	34	34
Number of constraints	35	37
CPU-time (DEC-20 computer)	28.30	30.77
% deviation from continuous solution		1.036 %

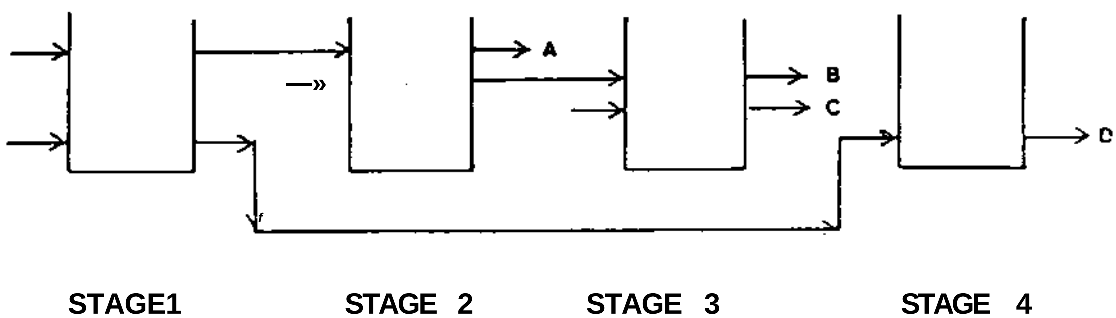


Figure 1« Plant for Example 1

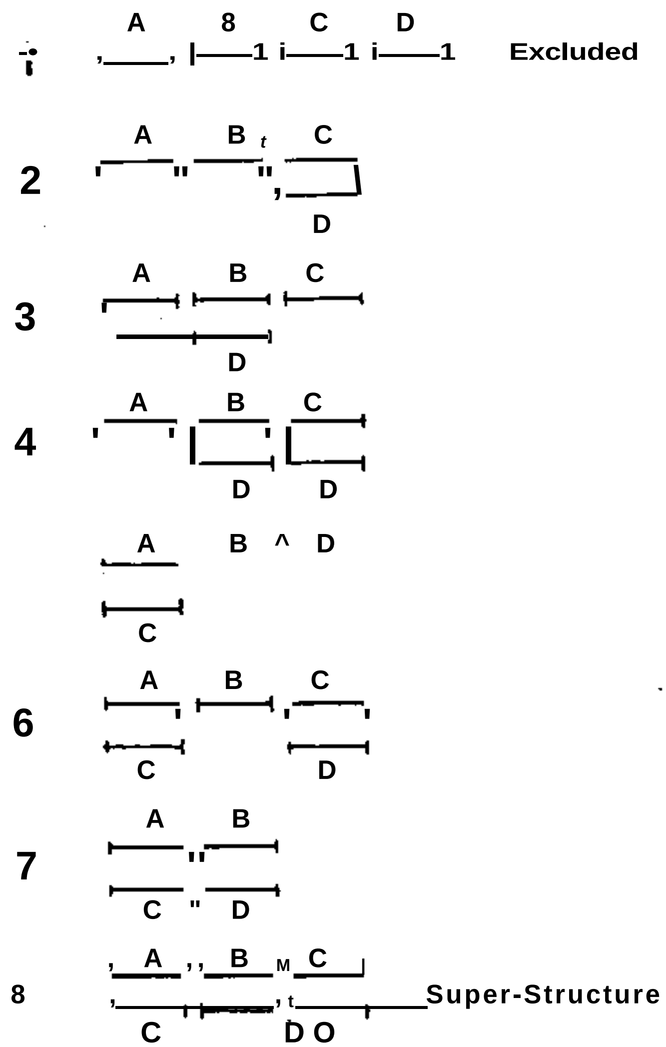


Figure 2. Schedule Alternatives for Example 1

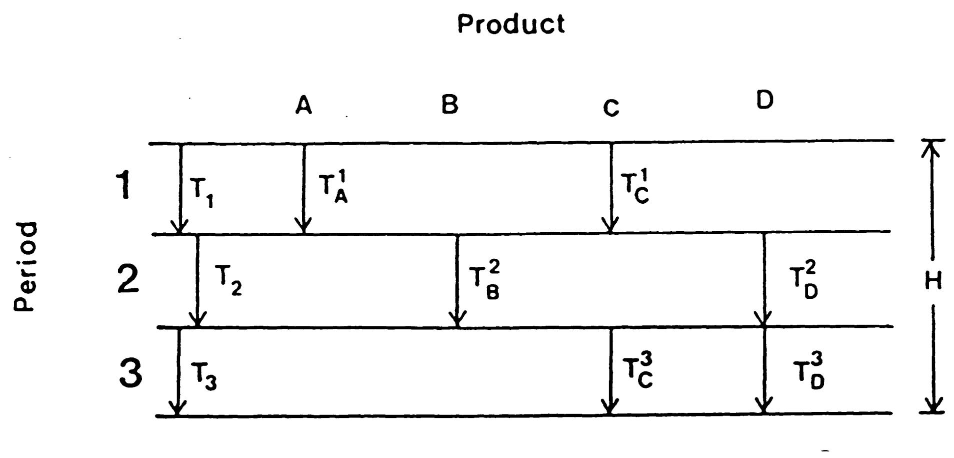


Figure 3. Super-Structure for Example 1

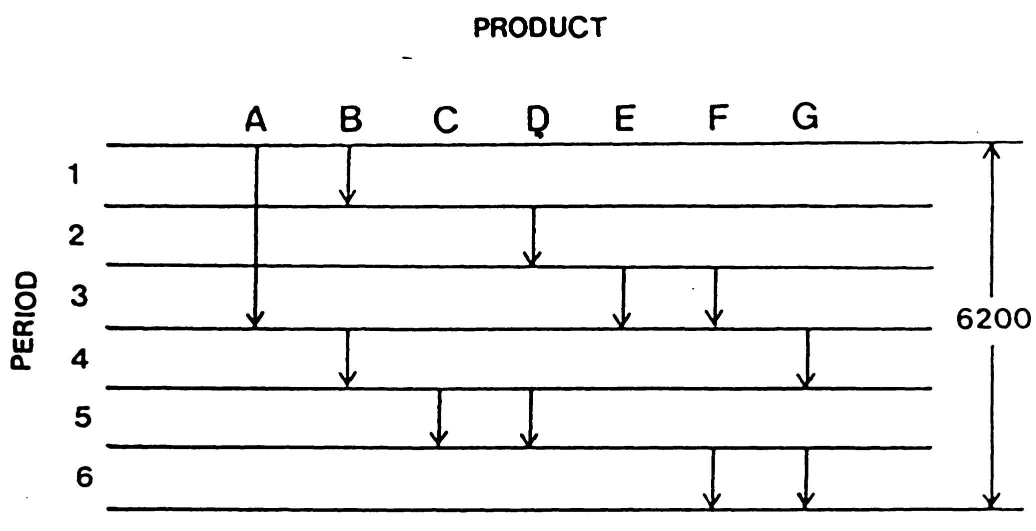


Figure 4. Super-Structure for Example of Suhami and Mah (1982)

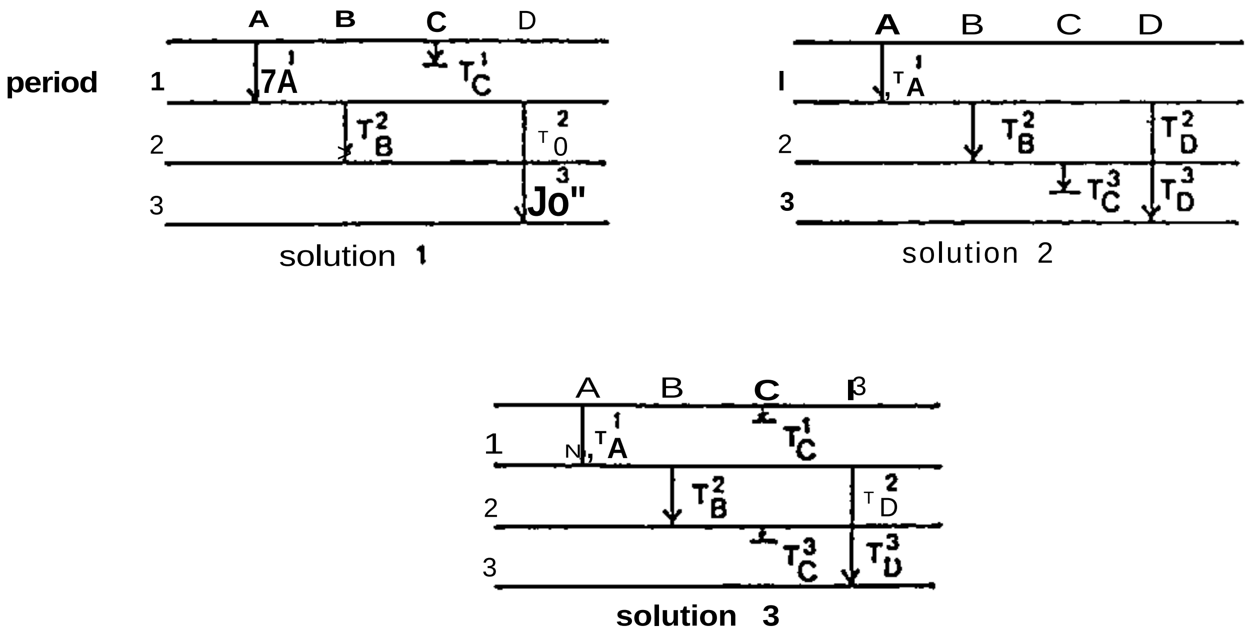


Figure 5. **Possible Solutions for Example 1**

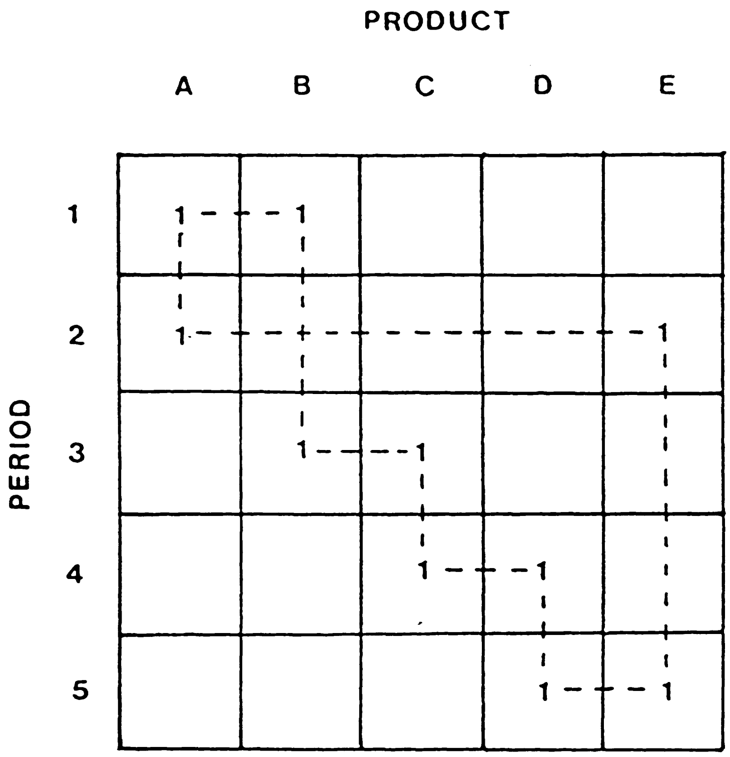


__Figure 6.__ Matrix Showing Loop of Cycle Example

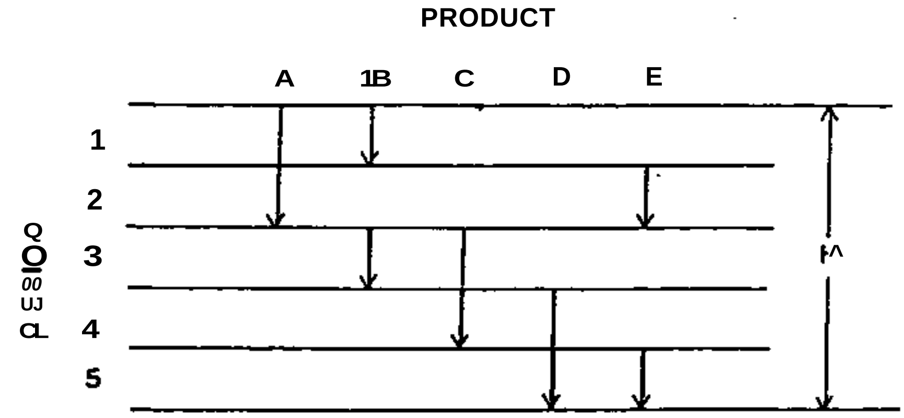


Figure 7. Super-Structure for Cycle Example

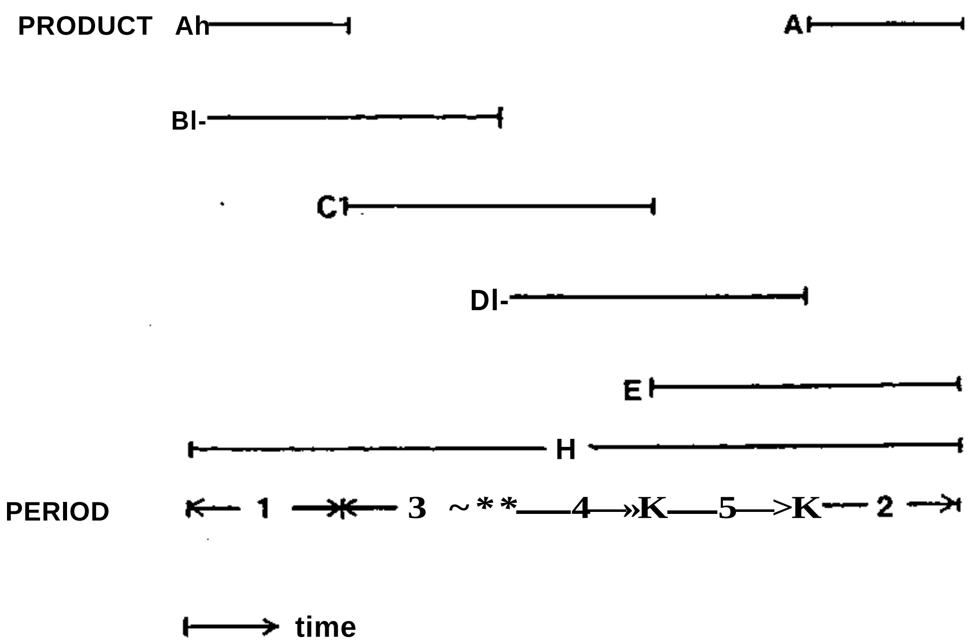


Figure 8. A Candidate Solution to Cycle Problem

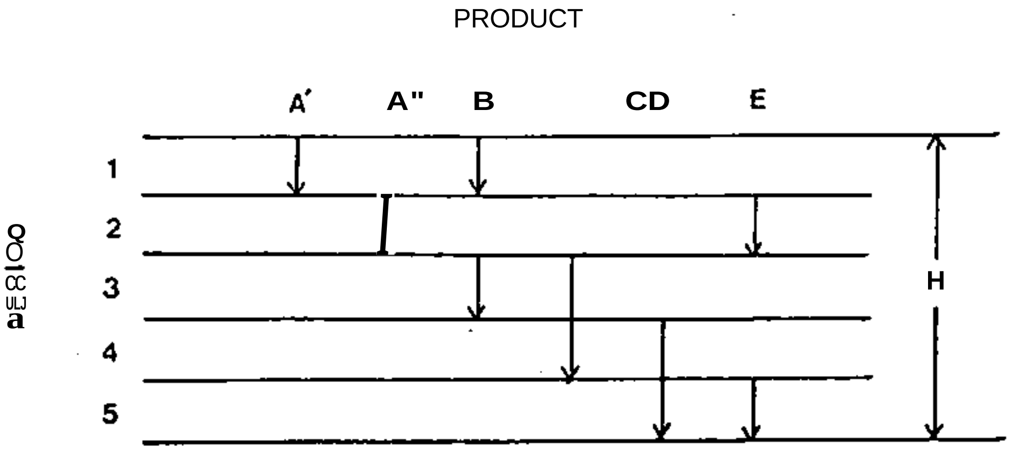


Figure 9. Super-Structure for Cycle Example with Products A' and A"

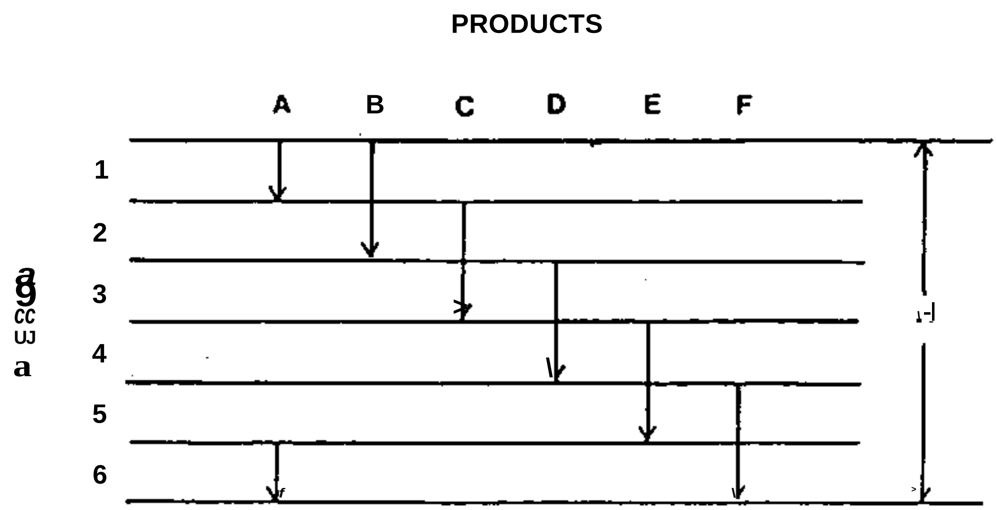


Figure 10. Super-Structure of a Six Product Example

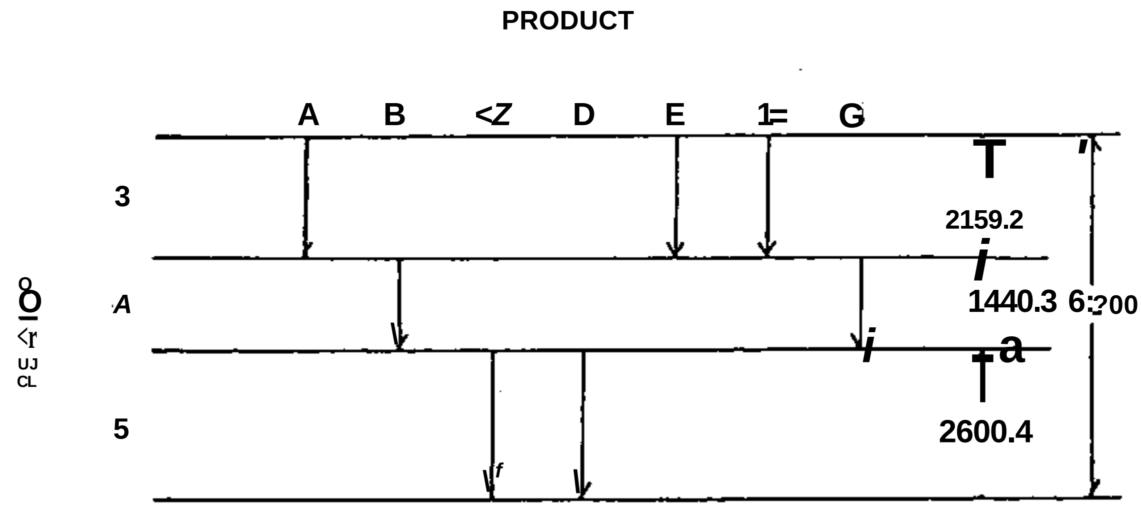


Figure 11. Optimal Schedule for Example 2

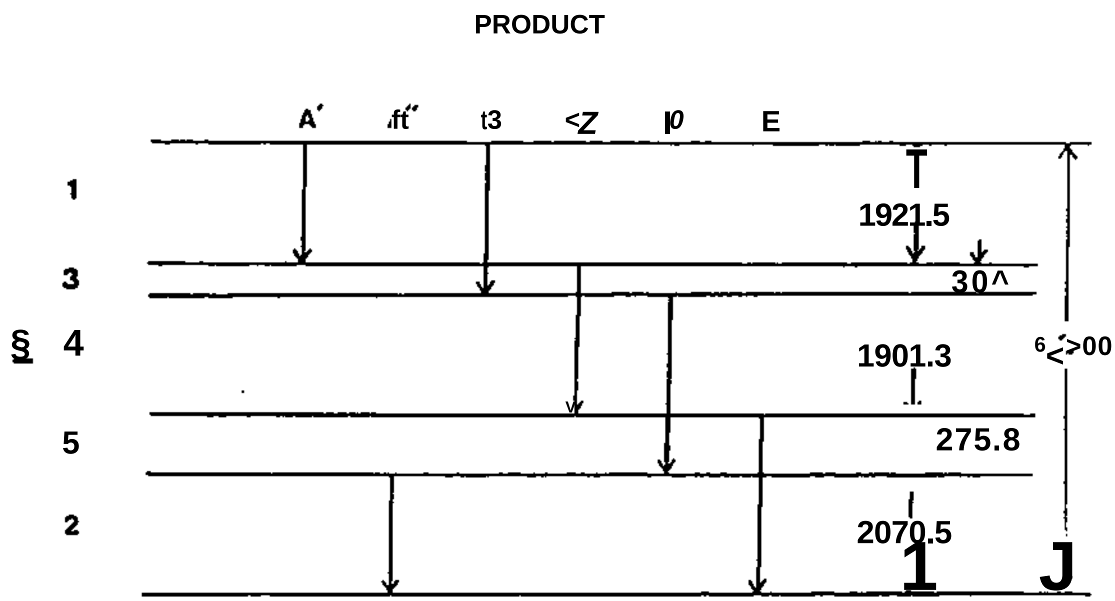
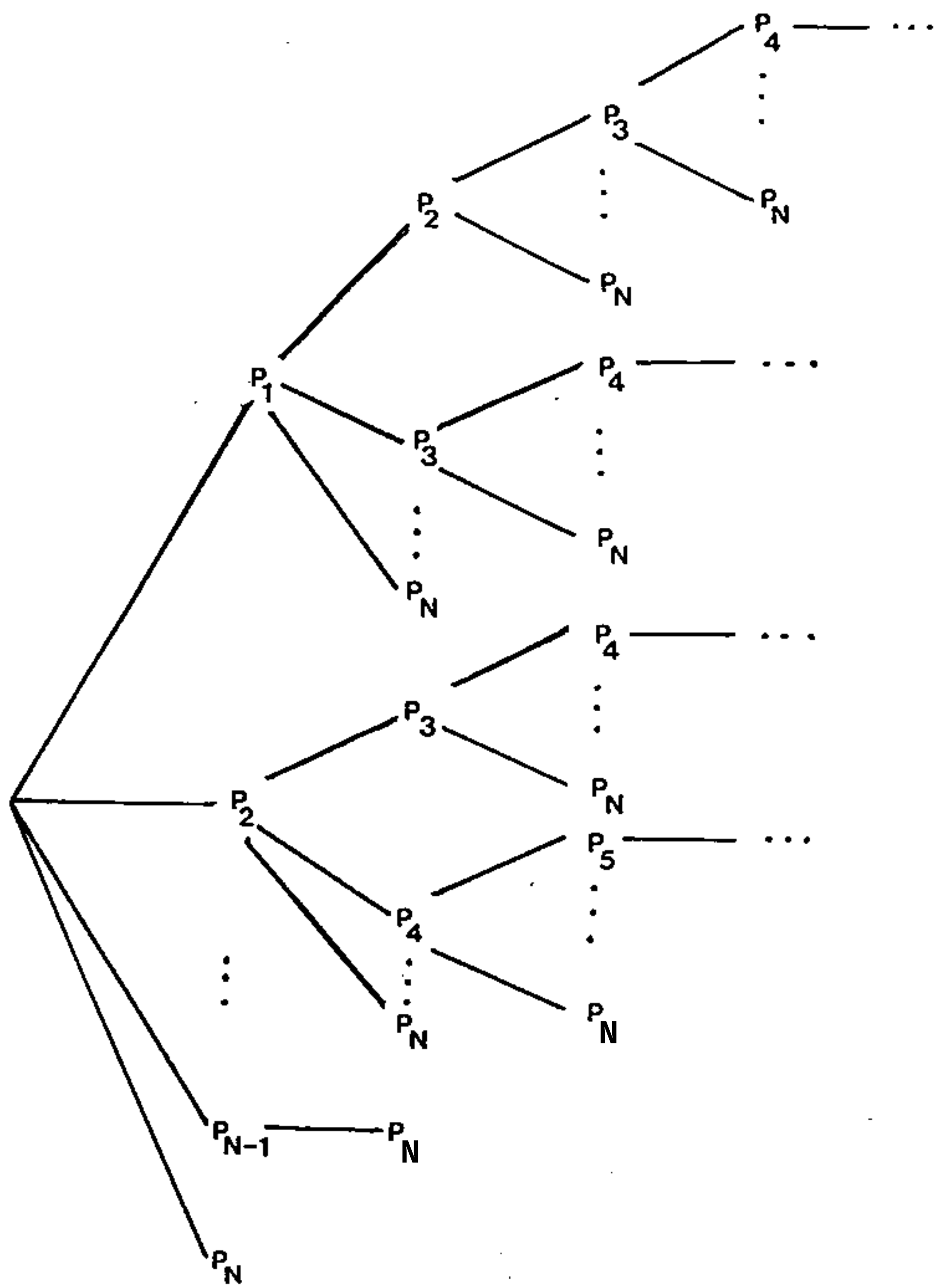


Figure 12. Optimal Schedule for Example 3



<u>Figure A-1.</u> Search Tree for Horizon Constraints

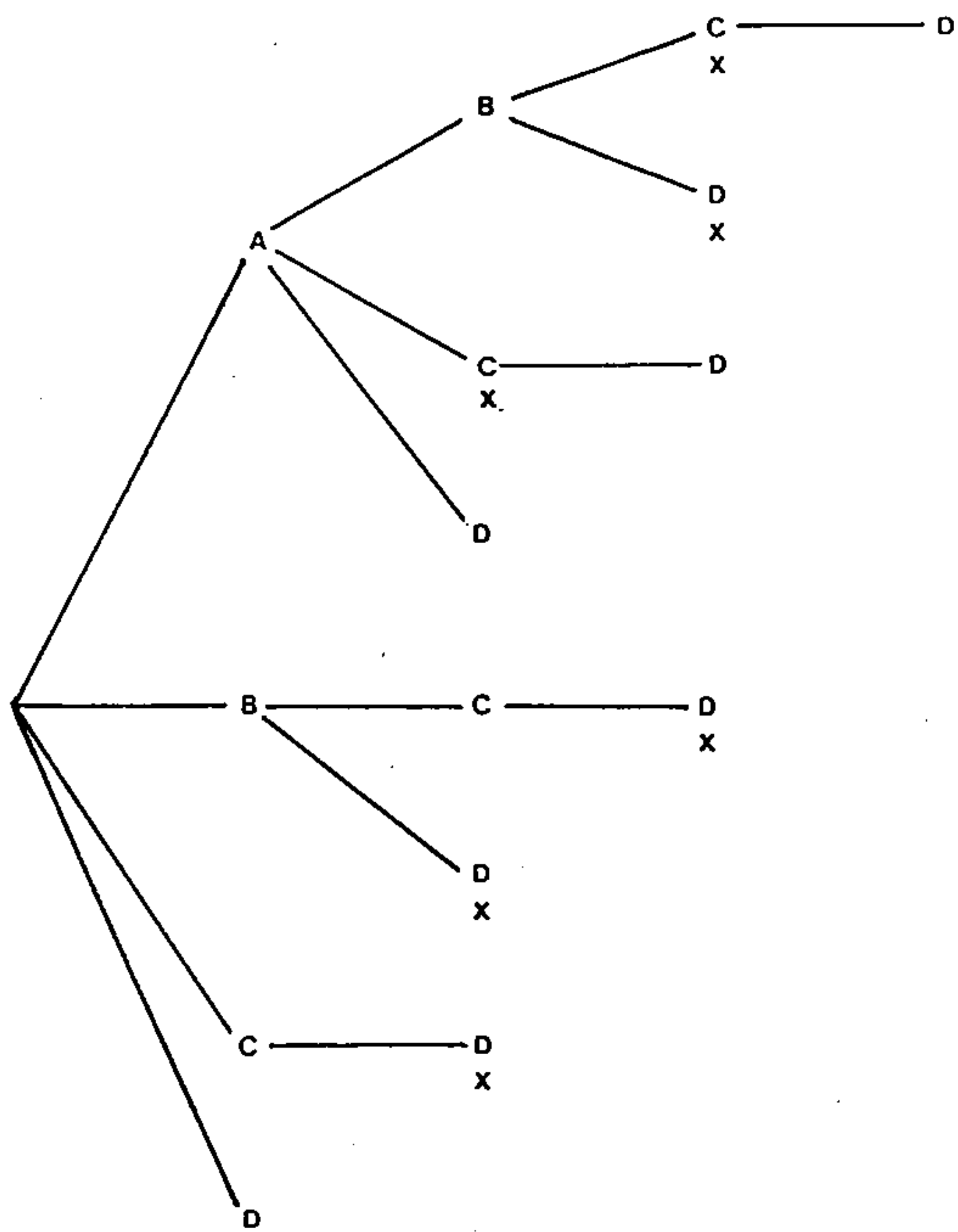


Figure A-2. **Search Tree** for Horlron Constraints of Example 1