Mixed-Integer Linear programming Feformulations for Batch Process Design

V.T. Voudouris, I.E. Grossmann

EDRC 06-99-91

Mixed-Integer Linear Programming Reformulations for Batch Process Design

V.T. Voudouris and I.E. Grossmann*
Department of Chemical Engineering
Carnegie Mellon University
Pittsburgh, PA 15213

August 1991

* Author to whom all correspondence should be addressed

Abstract.

The objective of this paper is to show that many nonlinear models for batch design, which are based on the assumption of continuous sizes, can be reformulated as MILP problems when sizes are restricted to discrete values. Problems considered include multiproduct plants operating with single product and mixed product campaigns, and multipurpose plants with single and multiple production routes. It is shown that by exploiting the structure of these MILP problems, solutions can be obtained with modest computational effort. In addition, as opposed to the use of rounding schemes for continuous models, global optimum solutions are guaranteed.

Introduction.

Recent growth in specialty chemicals, food products and pharmaceutical industries has increased the interest in systematic methods for the design of batch processes. Therefore, in recent years there have been increased research efforts to develop design methods and tools for multiproduct and multipurpose batch plants; see Reklaitis (1989) for a review.

Among the earlier works, the design problem for multiproduct batch plants operating with single product campaigns (SPC) was formulated as a nonlinear program by Robinson and Loonkar (1972). They used the minimum capital cost as a design criterion. The multiproduct model was extended by Sparrow et al (1975) to include the optimal selection of the number of parallel units at each processing stage as well as the optimization over discrete equipment sizes. They used heuristics as well as a specific branch and bound method for this problem. Grossmann and Sargent (1979) presented an alternative approach in which the problem by Sparrow et al was formulated as a mixed-integer-nonlinear program whose relaxation was shown to have a unique optimum.

The above design models were restricted to the case of single product campaigns with no intermediate storage and where only batch equipment were considered in the design procedure. Some of these simplifying assumptions were subsequently relaxed. One direction followed by some researchers was to take into account semicontinuous units (Knopf et al, 1981 , Yeh and Reklaitis, 1987). The other direction was the incorporation of scheduling with mixed product campaigns (Birewar and Grossmann, 1989).

The optimal design of multipurpose batch plants is more difficult than that of multiproduct plants. Initially the work concentrated on plants operating in SPC mode and single production routes (Suhami and Mah, 1982 , Vaselenak et al, 1987, Faqir and Karimi, 1988). Recently the assumption of single production routes has been relaxed to the more realistic case of plants with multiple production routes (Faqir and Karimi, 1990). Finally, a more complete formulation of the problem which also addresses the aspect of various different assignments of equipment types to tasks has been proposed by Papageorgaki and Reklaitis (1990). These authors considered further the case in which a given equipment type need not be exclusively dedicated to a single task, but the equipment items can be divided among multiple tasks of the same or different products.

Most of the models cited above correspond to nonlinear optimization problems (NLP or MINLP) in which the sizes of equipment are treated as continuous variables. In practice, however, it is clear that in virtually all the cases sizes are only available in discrete values. Therefore, there is currently this gap for the application in practice of optimization models for batch design .

The outline of this paper is as follows. Firstly we show that nonlinear models proposed previously in the literature, and which are based on the assumption of continuous sizes, can in fact be reformulated as MILP problems where the discrete sizes can be explicitly accounted for. Cases considered include multiproduct and multipurpose plants with single and multiple production routes. Methods for enhancing the solution efficiency of these models will also be discussed and illustrated with several example problems. The major significance of the MILP models presented in this work is that they correspond to realistic design models that can be solved to global optimality with reasonable computational expense.

Definitions and problem statement

Batch plants may be divided into two broad categories. First, the multi product batch plant in which the same sequence of equipment is used to produce each product. If a collection of process equipment is used in different arrangements to produce perhaps more than one product at any time then this corresponds to a multi purpose batch plant.

A multi product or multi purpose plant can be operated in two distinct modes. In the first mode, each product is produced in one campaign the single product campaign (SPC). Alternatively the plant can be operated with mixed product campaigns (MPC). In this mode two or more products are manufactured in each campaign and therefore the selection of the sequence of the products becomes a very important factor.

The staged nature of a batch plant, comprised of a number of units in series, allows four different storage options, i) Unlimited intermediate storage (UIS), ii) Finite intermediate storage (FIS), iii) No intermediate storage (NIS), iv) Zero wait or No wait (ZW or NW). In both the NIS and ZW modes, there is no storage between stages. After the completion of a batch in a processing unit, it may be held in it temporarily in the NIS mode, or in a storage vessel in the UIS or FIS modes. In the ZW mode the batch must be immediately transferred to the downstream unit. In situations where unstable intermediates are produced and must be immediately processed in subsequent steps, the ZW mode of operation is used. Therefore, the storage policy best describing batch plants is a combination of all the above modes, which will be referred to as mixed intermediate storage (MIS) policy.

The following notation will be used throughout the paper:

i	Index of products {A,B,C....} with cardinality N.
j	Index of stages {1,2,...., M} (index of equipment in problem'(P6)).
h	Index of production campaigns.
r	Index of routes in a multipurpose plant with multiple routes {1,2,...., R}.
s	Index of discrete sizes {1,2,...,, nsj}.
n	Index of number of parallel units {1,2,...., npj}.
tjj	Processing time of product i at stage j.
^{T}Li	Cycle time in single product campaigns, for product i. T^ = max{ty}.
Sjj	Size factor of stage j for product i.
Qi	Market demand for product i.
H	Time horizon in which the demand has to be satisfied.
Vj	Volume of a vessel at stage j.
VJ_S	Standard volume of size s for stage j.
Bi	Batch size for product i.
T[	Length of time which is dedicated to the production of product i.
θ_h	Length of campaign h.
a_{hi}	Incidence parameter denoting whether product i can be produced in campaign h (1 for positive case, 0 negative case),
cjj.	Amount of production from route r.
T_r	Length of time for production in route r.
n[	Number of batches of product i during the time horizon H.
NP_{ik}	Number of occurrences of the pair i-k in a MPC schedule during horizon H.

In this work we address the optimal design of a batch plant for which the objective is to minimize batch equipment cost with the equipment being available in discrete sizes. The plant in question consists of M stages of batch equipment that are to be used for producing N different products, and they can belong in any of the following cases:

1) Multiproduct plants operating under:

a) Single Product Campaigns with single unit per stage under ZW policy.
b) Single Product Campaigns with parallel units per stage under ZW policy.
c) Mixed Product Campaigns with single unit per stage under any storage policy.
d) Mixed Product Campaigns with parallel units per stage under UIS policy

2) Multipurpose plants operating with:
   a) Single production routes.
   b) Multiple production routes.

The basic data that are given are the product demands Qj and a time horizon H, the size factors Sjj (volume of a vessel in stage j required to produce one mass unit Df the final product) and the processing times tjj. The major assumptions made are the following. No splitting or mixing of a specific batch is allowed while in process. The processing time tjj of a product i in any stage j is independent of the batch size. The size factors Sjj are constant. Parallel units are allowed in each stage. Resource constraints are not taken into account except the availability of vessels in standard sizes.

Single product campaigns with single units per stage.

Consider the case of a multiproduct batch plant with one unit per stage operating with single product campaigns and ZW policy. The model proposed by Grossmann and Sargent (1979) for optimal sizing consists of the following NLP problem:

$$\min \sum_j a_j V_j^{b_j} \qquad \text{(CP1)}$$

$$\text{s.t.} \qquad V_j \geq S_{ij} B_i \qquad i=1,..,N \ , \ j=1,..,M$$

$$\sum_i \frac{Q_i}{B_i} T_{Li} \leq H$$

$$V_j^{lo} \leq V_j \leq V_j^{up} \qquad j=1,..,M$$

$$B_i \geq 0 \qquad i=1,..,N$$

where $T_{Li} = \max_{j=1,..M} \{t_{ij}\}$ is a constant as well as a_j (cost coefficient) and b_j (cost exponent).

This model is a geometric programming problem which involves posynomial terms, and therefore can be convexified with exponential transformations. A major assumption in this formulation is that the vessel sizes V_j are assumed to be continuous within specified lower and upper bounds V_j^{lo} ,V_j^{up}. In real life however, only vessels of discrete sizes are available. One alternative is to simply round up the solution of the NLP in (CP1),to discrete sizes although there is no guarantee that this will lead to a global optimal design. In order to rigorously tackle this situation the following discrete variables can be introduced:-

$$y_{js} = \begin{Bmatrix} 1 & \text{if unit at stage } j \text{ has size } s \\ 0 & \text{otherwise} \end{Bmatrix}$$

The variable V_j is restricted to take values from the set $SV_j=\{ v_{j1} , v_{j2} ,...,v_{jn} \}$. A straightforward extension of (CP1) to handle the case of discrete sizes is to introduce new constraints to restrict the volumes to discrete sizes. In this way the following MINLP model is obtained,

$$\min \sum_j a_j V_j^{b_j} \qquad \text{(P1)}$$

$$\text{s.t.} \quad V_j \geq S_{ij} B_i \qquad i=1,..,N \;,\; j=1,..,M$$

$$\sum_i \frac{Q_i}{B_i} T_{Li} \leq H$$

$$V_j = \sum_s v_{js} y_{js} \qquad j=1,..,M$$

$$\sum_s y_{js} = 1 \qquad j=1,..,M$$

$$V_j ,\; B_i \geq 0 \qquad i=1,..,N \;,\; j=1,..,M$$

$$y_{js} \in \{0 , 1\} \qquad j=1,..,M \;,\; s=1,..,ns_j$$

This model can be convexified with exponential transformations to find the optimal discrete sizes of the vessels (e.g. see Kocis and Grossmann, 1988). The drawback, however, is that the MINLP model may be large and expensive to solve.

In order to reformulate model (CP1) as an MILP we define $T_i = n_i T_{Li}$, the total time for producing product i where $n_i = Q_i / B_i$ is the number of batches of product i. We then have Qi / Bj = nj = *T[I* Ty . By multiplying the first constraint of (CP1) by Qi and after algebraic manipulations we get

$$T_i \geq \frac{Q_i S_{ij}}{V_j} T_{Li} \qquad (1)$$

The right-hand-side is nonlinear as it involves the inverse of the volume. But by taking advantage of the multiple choice character of the problem of discrete sizes, we can develop a model which is linear. First we note that 1/Vj can be expressed in terms of the discrete variables as:

$$\frac{1}{V_j} = \sum_s \frac{y_{js}}{v_{js}} \qquad (2)$$

$$\sum_s y_{js} = 1$$

$$y_{js} \in \{0, 1\}$$

In this way by substituting (2) into (1) and by setting CJs = aj $v_j^{\beta_j}$, the optimal design problem reduces to the MRP problem,

$$\min \sum_j \sum_s c_{js} y_{js} \qquad (\mathrm{Rpi})$$

s.t. $T_i \geq \sum_s \left(\frac{Q_i S_{ij}}{v_{js}}\right) y_{js} T_{Li}$ i=1,..,N , j=1,..,M

$$\sum_i T_i \leq H$$

$\sum_s y_{JS} = 1$ J=1,.,M

$T_i > 0$ i=1,..,N , $y_{js} \in \{0,1\}$ j=1,..,M , s=1,..,nsj

Note that this problem only has M additional constraints compared to the continuous model (CP1). However, potential problem of this formulation is the increase of the number of 0-1 variables when the number of discrete sizes is relatively high. This problem can be efficiently tackled by exploiting the specific structure of the problem as will be shown in the section of computational procedure. A more important aspect though, is that model (RP1), apart from being linear, can be solved for the global optimum combination of discrete sizes with the branch and bound method for MILP (see Nemhauser and Wolsey,19fc8). The continuous model (CPl), apart from being nonlinear, will in general not provide the global optimum with a simple rounding scheme.

Single Product Campaigns with parallel units*

When increased production rates are considered, it may be necessary to introduce units operating in parallel and out of phase to decrease the cycle times. These are defined by bottleneck stages which may be different for different products. As suggested by Kocis and Grossmann (1988), parallel units can be treated by introducing the discrete variable,

$$z_{jn} = \begin{cases} 1 & \text{if stage } j \text{ has n units of same volume in parallel} \\ 0 & \text{otherwise} \end{cases}$$

to represent an integer value for N_j the number of parallel units at stage j. Model (CPl) can then be expanded to treat the case of parallel units with the following MINLP model,

$$\min \sum_j N_j \alpha_j V_j^{\beta_j} \qquad \text{(CP2)}$$

$$\text{s.t.} \quad V_j \geq S_{ij} B_i \qquad i=1,..,N \;,\; j=1,..,M$$

$$TL_i \geq \frac{t_{ij}}{N_j} \qquad i=1,..,N \;,\; j=1,..,M$$

$$\sum_i \frac{Q_i}{B_i} TL_i \leq H$$

$$N_j = \sum_n n \, z_{jn} \qquad j=1,..,M$$

$$\sum_n z_{jn} = 1 \qquad j=1,..,M$$

$$V_j^{lo} \le V_j \le V_j^{up} \qquad j=1,..,M$$

$$z_{jn} \in \{0,1\} \quad , N_j , T_{Li} , B_i \ge 0 \qquad i=1,..,N \;,\; j=1,..,M \;,\; n=1,..,n_{Pj}$$

Again taking advantage of the multiple choice constraints for sizes and parallel units, it is possible to develop a linear model. Let

$$y_{jsn} = \begin{cases} 1 & \text{if stage j has n units in parallel of equal size s} \\ 0 & \text{otherwise} \end{cases}$$

By using a similar procedure as in the derivation of **(RP1), the MINLP** problem (CP2) can be reformulated as the MILP model,

$$\min \sum_j \sum_s \sum_n c_{jsn} \, y_{jsn} \qquad \textbf{(RP2)}$$

$$\text{s.t.} \qquad T_i \ge \sum_s \sum_n \left(\frac{Q_i S_{ij}}{v_{js}}\right)\left(\frac{t_{ij}}{n}\right) y_{jsn} \qquad i=1,..,N \;,\; j=1,..,M$$

$$\sum_i T_i \le H$$

$$\sum_s \sum_n y_{jsn} = 1 \qquad j=1,..,M$$

$$T_i \ge 0 \quad i=1,..N \;,\quad y_{jsn} \in \{0,1\}, \qquad i=1,..,N \quad j=1,..,M \quad n=1,..,n_{pj}$$

where $\quad c_{jsn} = n \, \alpha_j \, v_{js}^{\beta_j}$.

Mixed product campaigns with single units per stage.

In the case of mixed product campaigns it is assumed that a schedule may be allowed in which sequencing of products may be possible. To model this feature, special scheduling constraints have to be introduced as was shown by Birewar and Grossmann (1989). For the case of a plant with one unit per stage operating with ZW policy and assuming continuous sizes V_j, the following NLP model was proposed by these authors:

$$\min \sum_j a_j V_j^{b_j} \qquad \text{(CP3)}$$

s.t.

$$V_j \geq S_{ij} B_i \qquad i=1,..,N \ , \ j=1,..,M$$

$$n_i = \frac{Q_i}{B_i} \qquad i=1,..,N$$

$$\sum_k NP_{ik} = n_i \qquad i=1,..,N$$

$$\sum_i NP_{ik} = n_k \qquad k=1,..,N$$

$$\sum_i (n_i \ t_{ij} + (\sum_k NP_{ik} SL_{ikj})) \leq H \qquad j=1,..,M$$

$$NP_{ii} \leq n_i - 1 \qquad i=1,..,N$$

$$V_j^{lo} \leq V_j \leq V_j^{up} \qquad j=1,..,M$$

$$B_i , n_i \geq 0 \quad i=1,..,N \quad , \quad NP_{ik} \geq 0 \quad i=1,..,N \quad k=1,..,N$$

The third and fourth constraint of (CP3) are aggregated assignment constraints for product sequences. The sixth constraint of (CP3) is for eliminating subcycles that only involve one product. SL_{ikj} is the slack generated between products i and k at stage j if they are scheduled in that order and includes the cleanup time that may be required in order to prepare the vessel for product k after product i has finished processing. These slacks can be computed a priori as shown in Birewar and Grossmann (1989). In order to obtain an MILP design model it is required to reformulate the first constraint of (CP3) in terms of the number of batches due to the aggregated scheduling constraints. Since $n_i= Q_i / B_i$, the first constraint in (CP3) can be expressed as:

$$n_i \geq \frac{Q_i S_{ij}}{V_j} \qquad i=1,..,N \ , \ j=1,..,M \qquad (3)$$

Again, applying (2) and following a similar procedure as in the derivation of problem (RP1) the following MILP model is obtained,

$$\min \sum_j \sum_s c_{js}\, y_{js} \qquad \textbf{(RP3)}$$

s.t.

$$n_i \geq \sum_s \left(\frac{Q_i S_{ij}}{v_{js}}\right) y_{js} \qquad i=1,..,N \ , \ j=1,..,M$$

$$\sum_k NP_{ik} = n_i \qquad i=1,..,N$$

$$\sum_i NP_{ik} = n_k \qquad k=1,..,N$$

$$\sum_i \left(n_i t_{ij} + \left(\sum_k NP_{ik} SL_{ikj}\right)\right) \leq H \qquad j=1 \ldots M$$

$$NP_{ii} \leq n_i - 1 \qquad i=1,..,N$$

$$\sum_s y_{js} = 1 \qquad j=1,..,M$$

$$n_i, \ NP_{ik} \geq 0 \quad i=1,..,N \quad k=1,..,N \qquad y_{js} \in \{0,1\} \quad \forall j,s$$

The various storage policies can be addressed by assigning appropriate values to the slacks SL_{ikj}. For example the case of UIS storage policy can be treated by setting SL_{ikj} equal to the clean-up time for every i,k,j . Even further, in this case it is possible to address the case of parallel units per stage with the following MELP model,

$$\min \sum_j \sum_s \sum_n c_{jsn} \ y_{jsn} \qquad \textbf{(RP4)}$$

s.t.

$$T_i \geq \sum_s \sum_n \left(\frac{Q_i S_{ij}}{v_{js}}\right)\left(\frac{t_{ij}}{n}\right) y_{jsn} \qquad i=1,..,N \ , \ j=1,..,M$$

$$\sum_s \sum_n y_{jsn} = 1 \qquad j=1,..,M$$

$$\sum_k NP_{ik} = n_i \qquad i=1,..,N$$

$$\sum_i NP_{ik} = n_k \qquad k=1,..,N$$

$$\sum_i n_i t_{ij} \leq \sum_n \sum_s Hny_{jsn} \qquad J=1.».M$$

$$NP_{ii} \leq n_i - 1 \qquad i=1,..,N$$

$$q \ , \ NP_{ik} \geq 0 \quad \forall i,k \quad \text{and} \quad y_{jsn} \in \{0,1\} \quad \forall j.s.n$$

Multipurpose plants with single production routes.

The case considered here is the one where not all products require the same processing stages. Simultaneous production of products sharing the same stages in not allowed since only one production route is considered for each product with the possibility of parallel units (see Suhami and Mah, 1982). The model proposed for this problem by Vaselenak et al (1987), and later extended by Faqir and Karimi (1988), corresponds to the following MINLP:

$$\min \sum_j N_j a_j V_j^{\beta_j} \qquad \textbf{(CP5)}$$

s.t.

$$V_j \geq S_{ij} B_i \qquad i=1,..,N \ , \ j=1,..,M$$

$$T_{Li} \geq \frac{t_{ij}}{N_j} \qquad i=1,..,N \ , \ j=1,..,M$$

$$T_i B_i = Q_i T_{Li} \qquad i=1,..,N$$

$$f_k (T_1,T_2,.., V_.) \leq H \qquad k=1,..,K$$

$$V_j^{lo} \leq V_j \leq V_j^{up} \qquad j=1,..,M$$

$$T_{Li}, T_i, B_i \geq 0 \qquad i=1,..,N$$

$$N_j = \text{Integer} \qquad j=1,..,M$$

In the above model there are K horizon constraints, each one being a linear function of the time T_i dedicated to the production of a specific product i. Suhami and Mah (1982) first proposed the introduction of such constraints for the planning of specific groupings of products, A systematic way to obtain these constraints was proposed by Vaselenak et al (1987). Finally Faqir and Karimi (1988) addressed various ways to express these constraints. Model (CP5) is similar to (CP2) with the exception of the horizon constraints. Therefore, in analogy to (RP2) the following MILP model can be developed for the case of discrete sizes:

$$\min \sum_j \sum_s \sum_n c_{jsn} \, y_{jsn} \qquad \text{(RP5)}$$

s.t

$$T_i \geq \sum_s \sum_n \left(\frac{Q_i S_{ij}}{V_{js}}\right)\left(\frac{t_{ij}}{n}\right) y_{jsn} \qquad i=1,..,N \;,\; j=1,..,M$$

$$\sum_s \sum_n y_{jsn} = 1 \qquad j=1,..,M$$

$$\sum_h a_{hi} \Theta_h \geq T_i \qquad i=1,..,N$$

$$\sum_h \Theta_h \leq H$$

$$T_i \,,\; \Theta_h \geq 0 \qquad y_{jsn} \in \{0,1\} \quad \forall\, j,s,n$$

in which the third and fourth constraints are expressed in terms of campaign lengths, Θ_h, as proposed by Faqir and Karimi (1988). In these constraints a_{hi} represents the coefficient of an incidence matrix to denote whether or not product i corresponds to campaign h. The number and identity of possible campaigns of compatible products can be obtained with the procedure by Vaselenak et al , or alternatively, as described in Appendix A, through a sequence of maximal clique problems. As discussed above, previous authors have spent significant effort to express these constraints only in terms of the variables T_i as is the case in model (CP5). It is not straightforward, however, to express these constraints in terms of T_i. In the procedure by Faqir and Karimi (1988) this effort is quite large especially considering the fact that the computational benefits obtained when the third and fourth constraint in (RP5) are replaced by the ones expressed in terms of T_i are questionable. The number of the horizon constraints in (RP5) is equal to N+1 (N = number of products), whereas the number of reduced constraints in (CP5) is not specified and can well exceed N+1.

Multipurpose plants with multiple production routes.

The case of a plant with multiple production routes corresponds to the one where at a specific stage parallel units are allowed, but they operate independently giving rise to multiple production routes. Also for this reason the size of the vessels can be unequal, and consequently the batch sizes of the routes producing the same product can also be unequal. The definition of such a design model of the plant is more flexible and it can potentially provide cheaper designs than models for plants with single production routes.

The problem is formulated by assigning a maximum number of units in each stage (groups of equipment as seen in Fig 1). Each product i can then follow a different path inside the plant depending on the unit through which it is processed at the required stages (or groups). Each of the possible paths inside the plant is denoted by a route r. In this way a specific set of routes PR(i) is assigned to each product i. For every route r a specific subset of equipments ER(r) is assigned. The volume of every equipment V_j, j=1,..,NE , is allowed to take values from a set $SV(j)=\{v_{j1},v_{j2}, .. , v_{js},...,v_{jn(j)}\}$, where $v_{j1}=0$ to account fot the possibility for not selecting the given equipment. The batch size for every route is denoted as B_r, whereas total production of a route is q_r. It is assumed that the batch size is identical for all batches produced in a specific route. The total demand for a product that has to be satisfied in a specific time horizon H is Q_i. From the previous definitions it is clear that :

$$\sum_{r \in PR_i} q_r \geq Q_i \qquad i=1,..,N \tag{4}$$

The production is again assumed to occur in SPC campaigns with ZW policy. Every campaign h is the period of time in which one or more compatible routes which do not share the same equipment are producing the product assigned to each one of them. Again the term S_{rj} denotes the size factor of equipment j in route r, and T_{Lr} is the cycle time of route r.

Faqir and Karimi (1990) proposed for this case the following (MINLP) model.

$$\min \sum_j \sum_s c_{js} y_{js} \tag{P6}$$

$$\text{s.t.} \quad V_j \geq S_{rj} B_r \qquad \forall j \in ER(r),\ r=1,..,R$$

$$T_r B_r = q_r\, T_{Lr} \qquad r=1,..,R$$

$$V_j = \sum_s v_{js}\, y_{js} \qquad j=1,..,M$$

$$\sum_s y_{js} = 1 \qquad j=1,..,M$$

$$\sum_{r \in PR_i} q_r \geq Q_i \qquad i=1,..,N$$

$$f_k(T_1, T_2, .., T_p..) \leq H \qquad k=1,..,K$$

$$V_j \geq 0 \ \forall j, \qquad q_r, B_r, T_r \geq 0 \ \forall r, \quad y_{js} \in \{0,1\} \ \forall j, s$$

where $TL_r = \max_{j \in ER(r)} \{ t_{ij},\ r \in PR(i) \}$ is the cycle time.

Note that the last K constraints are the horizon constraints in terms of the lengths of productions T_i as discussed previously in the paper. The main difficulty in (P6) is with the second equation which is bilinear. This equation gives rise to nonconvexities and Faqir and Karimi proposed valid underestimators to avoid cutting off the global optimum with an MINLP algorithm. However, as will be shown below, problem (P6) can again be reformulated as an MILP.

By combining the first and second constraint in (P6) and taking into account the condition in (2) we get:

$$T_r \geq \sum_s \left(\frac{q_r S_{ij} T_{Lr}}{v_{js}} \right) y_{js} \qquad \forall j \in ER(r),\ r=1,..,R \tag{5}$$

In order to eliminate the bilinearities $q_r * y_{js}$ a new variable q'_{js} is defined to represent this crossproduct. The equivalence of the crossproduct and the new variable is forced with the following constraints (see also Grossmann et al, 1991):

$$q_r = \sum_s q'_{rjs} \qquad \forall j \in ER(r),\ r=1,..,R \qquad (6)$$

$$q'_{rjs} \le U_{rjs}\, y_{js} \qquad \forall j \in ER(r),\ r=1,..,R,\quad s=1,..,ns_j \qquad (7)$$

where U_{rjs} represents an upper bound for the variable q'_{rjs} and can easily be obtained analytically as will be illustrated later in equation (8). Making use of (5) and (7) the following MILP model is proposed,

$$\min \sum_j \sum_s c_{js} y_{js} \qquad \text{(RP6)}$$

s.t.

$$T_r \ge \sum_s \left([illegible] \right) \qquad \forall j \in ER(r),\ r=1,..,R$$

$$\sum_s y_{js} = 1 \qquad j=1,..,M$$

$$q_r = \sum_s q'_{rjs} \qquad \forall j \in ER(r),\ r=1,..,R$$

$$q'_{rjs} \le U_{rjs} Y_{js} \qquad \forall j \in ER(r),\ r=1,..,R,\ s=1,..,n_{sj}$$

$$\sum_{r \in ER_i} q_r \ge Q_i \qquad i=1,..,N$$

$$\sum_h a_{hr} \Theta_h \ge T_r \qquad r=1,..,R$$

$$\sum_h \Theta_h \le H$$

$$q'_{rjs},\ q_r,\ B_r,\ T_r,\ \Theta_h \ge 0 \quad \forall r,j,s,h \quad , \quad y_{js} \in \{0,1\}$$

Note that as in (RP5) the horizon constraints in (RP6) are expressed in terms of the campaign lengths Θ_h. The reason behind using these horizon constraints is the same as in the previous case of multipurpose plants with single production routes.

The upper bound Ujjs can be obtained analytically from the first constraint (RP6). Because of **the** second and fourth constraint in (RP6) only one entry in the summation of the first constraint in (RP6) will be nonzero. The upper bound for T_r is H. Hence, in the worst case **the** first inequality in (RP6) will be satisfied as an equality; therefore,'

$$U_{rjs} = i \; ? \; \overset{p}{\underset{lJ5}{s}} = \frac{C \quad VIS\ H}{T \quad \wedge r\ i\ ^{l}\ L\ r} \qquad V\ J\ G\ ER(r)\ ,\ r=1\_R,\quad s=1\_ns\ J \qquad (8)$$

Computational considerations.

While the proposed MILP models (RP1) to (RP6) have the important feature of handling discrete sizes and the capability of obtaining the global optimum, they can potentially become expensive to solve for large problems. Three aspects of the models allow us to enhance the efficiency of the computational procedure, namely : SOSl structures, bounding schemes for domain reduction and objective function cutoff.

SOS 1 structure.

A feature of the models that was exploited in linearizing the design models was the multiple choice structure of the models. Beale et al (1970) introduced the notion of Special Ordered Sets (SOS). The first kind of special ordered sets is that of type 1 (SOSl) (see also Williams (1985),Tomlin (1988)). This is a set of variables (continuous or integer) within which exactly one must be nonzero. This restriction can be treated by enforcing a special branching rule that recognizes that only one variable is nonzero. Therefore, great computational advantages can be gained, instead of treating them as a summation of binary variables.

In all the models that we proposed, there is an equation

$$\underset{s}{X} yjs = 1 \qquad \text{or} \qquad \underset{s\ n}{Il} yjsn = 1 \qquad (9)$$

This means that variables yj_s and yj_{sn} can be treated as special ordered sets. For every block of variables yj_s , ordered sets for each j are introduced.

In order to take computational advantage of the SOS 1 structure there must be an ordering in these variables as for instance given by increasing cost. The natural order of variables yj_s is given by increasing sizes which in turn corresponds to increasing costs. In the case of parallel units the natural order in the variables y_{jsn} is not necessarily preserved when the number of parallel units increases for the various sizes. Although in our implementation strict sequential ordering was not considered, SOS1 sets have the effect of reducing the nodes that are generated in the search tree. The favorable computational results gained by using SOS1 are illustrated in the results section.

Domain reduction.

Apart from using special ordered sets, computational enhancements can be obtained by reducing the domain of the binary variables. More specifically, the binary variable y_{js} is defined in a domain which corresponds to the crossproduct of the sets in which j and s take values. In some instances though we can derive analytical or computational lower bounds on the volumes v_j^{lo}. In these cases the inclusion of the discrete variables y_{js} for sizes $v_{js}<v_j^{lo}$ in the model has no useful purpose because they will lead to infeasible designs. Also, by not including these variables in the model, the number of the nodes in the branch and bound tree and the size of the model are reduced.

Analytical lower bounds can be obtained by taking explicitly into account the structure of the constraints. Specifically for the NLP model (CP1), which is the relaxation of (RP1), we have the following. The first constraint in (CP1) is forced to equality for a specific product i by a specific stage j which is the bottleneck stage for this product. If a stage was not a bottleneck for any product then the volume of the vessel at this stage could have been decreased without any effect in the objective function. Hence, the optimality condition requires that every stage is a bottleneck stage for at least one product. A lower bound for the volume of a unit at a specific stage can be obtained if this stage is assumed to be a bottleneck stage for every product. Since $T_i = (Q_i / B_i)\, T_{Li}$, by setting the first constraint as an equation in (CP1) we get:

$$T_i = \frac{Q_i\, S_{ij}}{v_j^{lo}} T_{Li} \qquad i=1,..,N \tag{10}$$

By requiring that the sum of the above variables be exactlly equal to the horizon time H, we get:

$$v_j^{lo} = \sum_i \left(\frac{Q_i\, S_{ij}}{H}\right) T_{Li} \tag{11}$$

This lower bound turns out to be fairly tight for the cases in which the scheduling policy is SPC with one unit per stage (model (RP1)). The way that the lower bounds were exploited is to avoid the introduction of a discrete variable denoting the existence of a size lower than the lower bound in the model. For the computational impact more is presented in the section of computational results.

For the case of parallel units per stage (model (RP2)), a straightforward extension of the previous procedure gives the following bound,

$$v^{lo}_{jn} = \sum_i \left(\frac{Q_i S_{ij}}{H\,n} \right) T_{Li} \qquad (12)$$

If for any n the bound v^{lo}_{jn} exceeds the upper bound of the volume V^{up}_j at a stage j then a lower bound of the number of parallel units n^{lo} can be set as the next highest integer for the ratio v^{lo}_{jn} / V^{up}_j. For model (RP5) the bound given in equation (12) can also be used since the horizon constraints in (RP2) are tighter than in (RP5). For the case of MPC scheduling policy (model (RP3) and (RP4)) a lower bound that can be obtained analytically is the same as in equation (11) and by assuming that all the slacks are zero. The inefficiency of these bound was demonstrated in our computational experience where the reduction of the domain of the binary variables was only marginal. In this case tighter lower bounds on the volumes must be obtained numerically. For example for problem (RP3) these can be obtained by solving the following LP to minimize the volume for each stage j (see (2)) in which the variables y_{js} are treated as continuous:

$$\max \quad \sum_s \frac{y_{js}}{v_{js}} \qquad (13)$$

$$\text{s.t} \qquad n_i \geq \sum_s \left(\frac{Q_i S_{ij}}{v_{js}} \right) y_{js} \qquad i=1,..,N \ , \ j=1,..,M$$

$$\sum_k NP_{ik} = n_i \qquad i=1,..,N$$

$$\sum_i NP_{ik} = n_k \qquad k=1,..,N$$

$$\sum_i \left(n_i \, t_{ij} + \left(\sum_k NP_{ik} \, SL_{ikj}\right)\right) \leq H \qquad j=1,..,M$$

$$NP_{i\,i} \leq n_i - 1 \qquad i=1,..,N$$

$$n_i\ ,\ NP_{ik}\ \geq 0 \quad \forall i,k \quad , \quad 0 \leq y_{js} \leq 1 \quad \forall j,s$$

The lower bounds are then given by:

$$V_j^{lo} = \frac{1}{\sum_s \frac{y_{js}}{V_{js}}} \qquad j=1,..,M \qquad (14)$$

The problems in (13) are LP's that are not very expensive to solve. It is expected that in larger models the benefit from the domain reduction that will be obtained by using lower bounds will compensate the extra computational efforts required to solve the LP's in (13). Of course the above scheme can be applied in all cases and for every model, but for the cases of SPC the analytical bounds proved to be satisfactory.

Since the branch and bound algorithm will tend to examine nodes for the lower cost designs, the derivation of upper bounds for volumes was found to have only a small effect. Again it is possible by combining analytical manipulations and computational practices to derive these upper bounds.

For the case of multipurpose plants with multiple routes, a simple way to obtain the lower bounds in volumes is by solving the following LP relaxation for every equipment j:

$$\min \sum_s v_{js} y_{js} \qquad (15)$$

s.t Constraints in (RP6)

Priority constraints for each equipment group

In **the** case of a multipurpose plant the plant is divided into groups of equipment of the same type. If all items in a group are allowed to take all sizes, then equivalent permutations of item sizes might be obtained because of the identical nature of the equipment. This can be avoided by prioritizing the items in each group with the following constraint $V_j \geq V_{j+1}$ for all equipment j and j+1 in the same group.

Objective function cutoff.

The most common method for solving MILP problem is the branch and bound (B&B) algorithm. Good lower and upper bounds in the objective function can greatly impact the efficiency of the method. A good upper bound can reduce the size of the B&B tree that needs to be enumerated. This bound is usually obtained using a heuristic procedure. One way to obtain this upper bound is to solve the relaxed LP in which the integrality of all the binary variables is removed. This then yields relaxed volumes V_{j} that can be computed for instance from (2). By rounding up these volumes we can obtain a feasible solution in which the binary variables have integral values. The objective value of this solution is used as an objective function cutoff.

For the case of the multipurpose plant with multiple routes the objective function cutoff obtained by simply rounding up the relaxed solution turns out to yield a weak upper bound. In order to obtain a good cutoff the following procedure is proposed.

Assume the following binary variable is defined:

$$z_r = \begin{cases} 1 & \text{if route } r \text{ exists} \\ 0 & \text{otherwise} \end{cases}$$

The following constraints can then be added to model (RP6):

$$z_r \le \sum_s y_{js} \qquad \forall j \in ER(r), \; r=1,..,R \qquad (16)$$

$$q_r \le Q_i z_r \qquad r=1,..,R, \; \forall i \in PR_r \qquad (17)$$

$$\sum_{r \in PR_i} z_r \ge 1 \qquad i=1,..,N \qquad (18)$$

The introduction of these constraints will increase the dimensionality of the problem and as such it seems a step in the wrong direction. The important thing though is that the space of routes has been introduced in which some logical manipulations can be performed.

Consider the plant which is given in Figure 1. Equipment group 2 and 4 have 3 equipment, whereas all the other equipment groups have 1 equipment. As far as the number of equipment per stage is concerned the solution of the problem is going to be one of the following nine alternatives:

	Group 1	Group2	Group3	Group4	Group5	Group6
1)	1	1	1	1	1	1
2)	1	2	1	1	1	1
3)	1	3	1	1	1	1
4)	1	1	1	2	1	1
5)	1	2	1	2	1	1
6)	1	3	1	2	1	1
7)	1	1	1	3	1	1
8)	1	2	1	3	1	1
9)	1	3	1	3	1	1

In order to avoid the effect of counting equivalent solutions and without loss of generality a priority in picking equipment can be introduced. So for example equipment 2 in Group 2 should be picked before equipment 3 and so forth. An immediate result of the above is that for each one of the alternatives presented above, a significant number of binary variables z_r can be fixed. As an example the first alternative can be represented by fixing the binary variables z_r as follows: $z_1 = z_4 = z_7 = z_{10} = 1$, and $z_r = 0$ for all others.

In the second alternative we have two equipment in group 2 and 1 in group 4. For the first statement and keeping in mind the ordering of the equipment we can introduce the logical constraint which states that either route 1 or route2 and either route 10 or route 11 have to be picked whereas for the second statement we can fix the variables $z_7 = z_4 = 1$. All the other variables are fixed to zero. As we continue to the next alternatives, fewer variables are fixed and more variables are placed in exclusions. At the ninth alternative we are not able to fix any variable and we just obtain constraint (18).

The sequence of alternatives has some interesting characteristics. First in the initial alternatives the number of variables that are fixed is large. This has the effect that through constraints (16),(17) and (18) the problem is simplified significantly. The second interesting characteristic is that by following the sequence of plant configurations with an increasing number of units per group, it is quite likely that a good feasible solution can be obtained with little computational effort. For the example in Fig.1 a feasible solution with a cost of \$134,400 was obtained for the first alternative after 5 sec of CPU time in a Vax-6420. The optimal solution with a cost of \$124,500 was obtained in the second alternative (2 units in group 2, 1 unit in the other groups). The biggest problem was to prove the optimality of this solution by solving the subsequent problems. It is worth to note that using the method of rounding up the relaxed solution of the MELP in model (RP6) gave an upper bound in the capital investment of $188,400, which was unsatisfactory.

The upper bound of the cost described above can also be used to obtain upper bounds for the volumes of each equipment. This is done by solving the following LP relaxation for each equipment j:

$$\max \sum_{s} \wedge s\, y_{js} \tag{19}$$

s.t Constraints in (RP6)

Priority constraints for each equipment group

$$\sum_{j} \sum_{s} c_{js}\, y_{js} \leq cost^{u}$$

where CJ_S represents the cost of equipment j when it has a size s.

The techniques described above are not the only ones with which the computational efficiency of the MILPs can be enhanced. Another technique is to introduce cutting planes which will cut a part of the feasible region of the relaxed LP without eliminating integer solutions and if possibly to be facets of the convex hull of the integer problem. For more details about such techniques see Nemhauser and Wolsey (1988).

Computational results.

Example 1.

Rounding the solution of continuous NLP models reported in the literature has the drawback of giving suboptimal solutions. For example consider the case of a multiproduct plant with one unit per stage operating under the SPC/ZW policy. The plant consists of 6 stages and is dedicated to the production of 5 products A, B, C, D and E. Data for this problem are given in Table I. One way to solve the problem is using model (CP1). The corresponding NLP of this problem has 32 constraints and 12 variables. Minos 5.2 required 1.25 CPU sec and 44 iterations to obtain the optimal solution with a cost of $2,314,896. GAMS 2.25 on a Vax-6420 was used in order to generate the models. The optimal sizes of the vessels predicted are Vi=6017.59 , V2=3483.6 , V3=3960.9 , V4=4823.5 , V5=4646.5 , V6=3885.55 (in liters). Assume, however, that the vessels are only available in the following set of discrete values SV={3000, 3750, 4688, 5860,7325} liters. Note that the ratio of two consecutive sizes is constant and in this case this ratio is 1.25. Off course this ratio is arbitrary. In real applications the formulas or tables proposed by DIN or ANSI norms, are used in order to calculate the set of discrete sizes. By rounding up the NLP solution we get Vi=7325 , V2=3750 , V3=4688, V4=5860, V_5=4688, V6=4688 liters and a cost of $ 2,521,097.

Using the MILP model (RP1) the availability of discrete sizes is taken explicitly into account. The solution in this case is Vi=5860 , V2=3750 , V3=3750, V4=5860, V_5=4688, V£=4688 liters with a cost of $2,405,840 which is $115,257 cheaper or 4.8% lower than the previous value. It is clear that the rounding scheme did not predict the global optimal design.

Model (RP1) is an MILP which is solved using SCICONIC 2.11 through GAMS 2.25 in a Vax-6420. The MILP problem involves 38 constraints, 36 variables and 246 nonzero elements. The time required for the MILP was 2.95 CPU sees. If instead model (PI) is used, which is an MINLP, the problem involves 45 constraints and 43 variables with 140 nonzero elements; 30 of the variables are discrete. The computer code DICOPT++ required 12.18 CPU sees to solve this problem and obtained the same solution as the MILP. Note that the MILP is smaller in size than the MINLP in model (PI).

In order to illustrate the effect that the number of discrete sizes has in the size of model (RP1) as well as in the computational performance, three more cases, one with 8 , one with 15 and another with 29 discrete sizes were considered. The comparison is presented in Table II. We note from Table II that the number of discrete sizes has a significant effect in the number of 0-1 variables, and hence in the number of iterations and the CPU time. The way that the MILP's are treated so far is without using any of the techniques proposed in the previous section (SOS1, domain reduction , cutoff).

In Table III results are presented for the same problem but with the use of SOS 1, domain reduction and objective function cutoff. Note that the bounds are affecting mainly the size of the model whereas the SOS1 and cutoff affect the iterations and the CPU time. From the results it is seen that the handling of standard sizes gives rise to rigorous and robust models which can also be solved quite efficiently. Even when the number of discrete sizes is quite large the models that we propose can be solved in reasonable computational time.

Example 2.

The issue of computational efficiency becomes even more critical in the case of multiproduct batch plants with parallel units. Consider the problem given in Table IV. Note that the data are exactly the same as in Table I with the only difference in the processing time of product G in stage 4 which was increased significantly in order to bottleneck this stage and to introduce the need of parallel units. It is assumed that all the stages can have up to 4 units in parallel. In order to solve this problem the MINLP model (CP2) for continuous sizes and the MILP model (RP2) for discrete sizes were considered. For the MILP two cases have been solved. The first one involving 4 discrete sizes in the range between 1000 and 2500 liters, and in the second case 14 discrete sizes in the same range. The optimal cost predicted in the first case was $5,261,290. whereas in the second case $5,107,530. The results concerning the performance of the models with SCICONIC 2.11 are shown in Table V. From this table it is obvious the significant effect that the SOS1 structure and the domain reduction has.

Example 3.

For the case of plants with MPC and ZW policy the data for the example problem are the same as in Table I. As far as the number of discrete sizes are concerned two cases have been considered. In the first case the five discrete sizes of set SV were assumed, whereas in the second case 15 discrete sizes ranging from 3000 to 6500 liters with an interval of 500 liters were assumed. The optimal investment in the first case was $2,405,840 and in the second case $2,331,240. The computational performance using SCICONIC 2.11 is shown in Table VI. In both cases the cleanup times were assumed to be zero.

Example 4.

For the case of a multipurpose plant with single product campaigns and single production routes, the problem given in Yaselenak et al (1987) has been used. This problem involves 10 stages and 7 products. In each stage up to 3 parallel equipment are allowed. The solution in Vaselenak et al involved a capital investment of $355,516 and the continuous solution was obtained using MINOS/Augmented in 18.79 seconds in a DEC-20. The size of the model was 69 constraints and 48 variables. Assume that the volumes can take discrete values from the following set SV={1000, 1600, 2560, 4096, 6554, 10485}. If the continuous solution is rounded to the standard sizes then the capital investment is $397,457. Using the MILP model (RP5) the solution that was obtained required a capital investment of $387,695 representing an improvement of 2.51% in the capital investment. The size of the model was 42 constraints, 194 variables of which 180 were 0-1 variables and 824 nonzero elements. The solution was obtained in 21.07 seconds in a Vax-6420 using SCICONIC 2.11 through GAMS 2.25 and required 789 iterations. If SOS1 and domain reduction are used then the problem involves 42 constraints, 159 variables, 655 nonzero elements and required 5.51 seconds and 187 iterations.

Example 5.

For the case of a multipurpose plant with multiple production routes the example of Faqir and Karimi (1990) has been used. The data for this problem are given in Table VII whereas the layout of the plant is shown in Figure 1. The solution that the above authors obtained is a capital investment of $124,500. As far as the computational requirements of the procedure they devised, no clear characteristics can be given mainly because the procedure is highly interactive. If model (RP6) is used without any domain reduction, without use of SOS1 and without objective function cutoff then the model involved 50 0-1 variables, 245 variables (both continuous and binary), 859 nonzero elements and 238 constraints.The reason that in this case 50 binary variables were used is that no 0-1 variables were introduced for zero volumes, and the second constraint in (RP6) was relaxed as an inequality. Using SCICONIC 2.11 through GAMS2.25 , 329.42 CPU seconds on a Vax-6420 and 16,608 iterations were required to solve the problem. By formulating the problem with additional 0-1 variables for zero volumes in order to have the second constraint in (RP6) as an equality and using the SOS1 structure, the problem involved 60 binary variables, 238 constraints, 255 variables (continuous and binary), 869 nonzero elements. Using SCICONIC 2.11, 139.17 CPU seconds and 6923 iterations were required in order to obtain the solution. Using SOS1, domain reduction and objective function cutoff ($134,400 as discussed previously in the paper) the model had a size of 236 constraints, 246 variables and 853 nonzero elements. The CPU time required was 30.64 seconds and the number of iterations was 1572 on the same computer. Faqir and Karimi used a different model to solve the same problem which was nonlinear and nonconvex. They also proposed a somewhat complicated preprocessing scheme where domain reduction was also considered. Using the results of their domain reduction in our model and using SOS1 as well as the objective function cutoff the model involves 160 constraints, 144 continuous variables and 497 nonzero elements. The CPU time required to solve the model is 4.26 seconds in a Vax-6420, and 289 iterations are needed.

Conclusions.

It has been shown in this paper that taking into consideration the availability of the vessels in only discrete sizes, it is possible to reformulate existing nonlinear design models for batch processes as MILP problems. The resulting models determine the global optimal solution for the design of batch plants either multiproduct or multipurpose, and operating under various modes. Furthermore, the nonlinearities of the existing models are eliminated giving rise to more robust computations.

Although the resulting models can be solved without any preprocessing, significant computational gains can be achieved using the specific structure of the MILP's. The use of objective function cutoff, and especially the domain reduction using a bounding scheme, as well as the use of Special Ordered Sets of type 1 (SOS1) proved to have a significant impact in the performance of the solution procedure.

Finally, it should be noted that the reformulation approach presented in this paper for converting nonlinear optimization problems with discrete sizes as MILP problems has been generalized and applied to other problems as discussed in Grossmann, Voudouris and Ghattas (1991).

Acknowledgement.

The authors gratefully acknowledge financial support from the National Science Foundation under Grant CBT-8908735.

References.

Beale E.M.L, Tomlin J.A. "Special facilities in a general mathematical programming system for non-convex problems with ordered sets and **variables", *Proceedings of the Fifth International Conference on Operational Research,*** Lawrence J. editor,Tavistock Publication^ London, 1970, pp447-454

Birewar D.B, Grossmann LE, "Incorporating scheduling in tha optimal design of multiproduct plants", *Comp&Chem.Eng.yoX* 13, No 1/2, ppl41-161, 1989

Brooke A., Kendrick D., Meeraus A., "GAMS A Users Guide", The Scientific Press , 1988

Faqir N.M,Karimi LA, "Design of multipurpose batch plants with multiple production routes", *Proceedings FOCAPD'89,* Snowmass Village CO, pp 451-468, 1990

Faqir N.M,Karimi LA," Optimal design of batch plants with single production routes", Paper 79a, Design&Analysis II, AIChE Annual Meeting, Washington 1988

Grossmann LE, Sargent R.W.H, " Optimum design of multipurpose chemical plants ", *Ind.Eng.Chem.Proc.Des.Dev.* 1979, Vol 18, Nd2, pp343-348

Grossmann LE, Voudouris V.T., Ghattas O."Mixed-Integer Linear Programming Reformulation for Some Nonlinear Discrete Design **Optimization Problems", *Proceedings International Conference on Recent*** *Advances in Global Optimization,* Princeton Univ, May 10-11,1991

Knopf F.C,Okos M.R,RekIaitis G.V, "Optimal design of Batch/semicontinuous processes", *Ind.Eng.Chem.Proc.Des.Dev.* 1982, 21, pp79-86

Kocis G.R., Grossmann LE.,"Global optimization of nonconvex MINLP problems in process synthesis", *Ind.Engng.Chem.Res.* 27, pp 1407-1421, 1988.

Nemhauser G.L, Wolsey L.A. " Integer and Combinatorial Optimization",Wiley, 1988

Papadimitriou C.H,Steiglitz K. /'Combinatorial optimization: Algorithms and complexity", Prentice Hall, 1982

Papageorgaki S, Reklaitis G.V "Optimal design of multipurpose batch plants-1. Problem formulation ", *Ind.Eng.Chem.Res.* 1990a, 29, pp 2054-2062

Reklaitis G.V "Progress and issues in computer-aided batch process design", FOCAPD Proceedings, Elsevier, NY, 1990, pp 241-275

Robinson J.D,Loonkar Y.R , "Minimizing capital investment for multi-product batch plants", *Process Technology Intl,* 1972 , 17 , No 11 , p 861

SCICONIC/VM 2.11 " Users Guide", Scicon Ltd, 1991

Sparrow R.E,Forder G.J,Rippin D.W.T, " The choice of equipment sizes for multiproduct batch plant. Heuristic vs. branch and bound", *Ind.Eng. Chem.Proc.Des.Dev.* 1975, No3, Vol 14, pl97

Suhami I,Mah R.S.H, "Optimal design of multipurpose batch plants", ***Ind.Eng.Chem.Proc.Des.Dev.* 1982, 21, pp 94-100**

Tomlin J.A. "Special ordered sets and an application to gas supply operations planning", *Mathematical Programming,* 42,1988, pp 69-84

Vaselenak J.A **9Grossmann LE,Westerberg** A.W, " An embedding formulation for the optimal scheduling and design of multipurpose batch plants", *Ind.Eng.Chem.Res,Vol* 26, Nol,1987,ppl39-148

Williams H.P "Model Building in Mathematical Programming", second edition, Wiley, NY, 1985

Yeh N.C.,Reklaitis G.V, " Synthesis and sizing of batch/semicontinuous processes", *Comp&Chem.Eng.,Vol* 11,No 6,pp639-654,1987

Appendix A. Determination of campaigns for multipurpose plants

Consider the case where five products A,B,C,D and E are going to be produced in a multipurpose plant as shown in Figure A1. A graph G=(V,E) can be constructed where each node represents a product and an edge represents compatibility between two products. By compatibility it is meant that the production routes of the two products do not share a common equipment and can thus be processed in the same campaign. For our case the graph is shown in Figure A2. The different possible campaigns of compatible products is equivalent to a sequence of maximal clique problems as shown below.

A clique of a graph is defined as being a subset C of V in which all nodes are connected to each other. A clique is maximal if it is not a subset of any other clique in a graph. In Figure A3 the maximal cliques of the graph G=(V,E) are displayed. Here each clique represents a possible production campaign (i.e. produce A and D, C and D, A and D, and B-C-E simultaneously). Using the clique representation, a clique matrix can be generated in which the rows represent a clique and the columns represent a product. This clique matrix is the matrix A with entries a_{hi} that is used in models (RP5) and (RP6).

As discussed in Papadimitriou and Steiglitz (1982) the problem of finding the maximal cliques of a graph is a well studied problem, although no polynomial algorithm has been developed.

TABLE I: Data for example 1 (SPC with one unit per stage).

	Size factor Sij (l/kg)					Proc. time tij (h)					Cost coeff. ct(j) ($)	Costexp B(i)
	A	B	C	D	E	A	B	C	D	E		
Stage 1	7.9	0.7	0.7	4.7	1.2	6.4	6.8	1	3.2	2.1	2500	0.6
Stage 2	2	0.8	2.6	2.3	3.6	4.7	6.4	6.3	3	2.5	2500	0.6
Stage 3	5.2	0.9	1.6	1.6	2.4	8.3	6.5	5.4	3.5	4.2	2500	0.6
Stage 4	4.9	3.4	3.6	2.7	4.5	3.9	4.4	11.9	3.3	3.6	2500	0.6
Stage 5	6.1	2.1	3.2	1.2	1.6	2.1	2.3	5.7	2.8	3.7	2500	0.6
Stage 6	4.2	2.5	2.9	2.5	2.1	1.2	32	6.2	3.4	22	2500	- 0.6

Q(A)= 250000, Q(B)=150000, Q(C)=180000, Q(D)=160000, Q(E)=120000 (KQ)

TABLE II: Computational results on example 1 (Without SOS 1,domain reduction and cutoff).

# dis. sizes	Constr.	All Variables	Nonzeros	0-1 var's	CPU *	Iter.
5	38	36	246	30	3.1	195
8	38	54	372	48	2.93	181
15	38	96	666	90	25.09	985
29	38	180	1254	174	44.94	1203

* In Vax-6420 sees

TABLE HI: Computational results on example 1

(With SOS 1,domain reduction and cutoff).

# dis. sizes	Constr.	All Variables	Nonzeros	0-1 var's	CPU *	Iter.
8	38	46	316	40	1.93	57
15	38	82	568	76	2.85	91
29	38	154	1072	148	6.64	182

*** In Vax-6420 sees**

Table IV: Data for example 2 (SPC with parallel units per stage).

	Size factor Sij (l/kg)					Process, time tij (h)					Cost coeff.	Cost exp
	A	B	C	D	E	A	B	C	D	E	a(t) ($)	0(0
Stage 1	7.9	0.7	0.7	4.7	1.2	6.4	6.8	1	3.2	2.1	2500	0.6
Stage 2	2	0.8	2.6	2.3	3.6	4.7	6.4	6.3	3	2.5	2500	0.6
Stage 3	5.2	0.9	1.6	1.6	2.4	8.3	6.5	5.4	3.5	4.2	2500	0.6
Stage 4	4.9	3.4	3.6	2.7	4.5	3.9	4.4	41.9	3.3	3.6	2500	0.6
Stage 5	6.1	2.1	3.2	1.2	1.6	2.1	2.3	5.7	2.8	3.7	2500	0.6
Stage 6	4.2	2.5	2.9	2.5	2.1	1.2	3.2	6.2	3.4	12	2500	0.6

Q(A)= 250000, Q(B)= 150000, Q(C)=180000, Q(D)=160000, Q(E)=120000 Kg

TABLE V: Computational results for SPC with parallel units per stage.

# dis. sizes	Constr.	All Variables	Nonzeros	0-1 var's	CPU *	Iter.
4	38	102	708	96	2.28	166
4 **	38	37	253	31	2.33	104
14	38	342	2388	336	37.010	1124
14 **	38	240	1674	234	8.35	374

" SOS1 and domain reduction have beerl used. * In Vax-6420 sees

TABLE VI: Computational results for MPC/ZW with one unit per stage.

# dis. sizes	Constr.	All Variables	Nonzeros	0-1 var's	CPU *	Iter.
5	53	61	456	30	4.54	238
5 *•	53	52	393	21	1.32	97
15	53	121	876	90	39.51	1096
15 "	53	99	722	68	5.1	259

** SOS1 and domain reduction have beeri used. * In Vax-6420 sees

TABLE VII. Data for the multipurpose plant with multiple production routes.

Product	Equipment Requirement Task 1	Task 2	Task 3	Processing Times h Task 1	Task 2	Task 3	Size Factors L/kg Task 1	Task 2	Task 3	Production Requirement kg
A	Group 1	Group 2	-	4.0	6.5	-	1.5	2.0	-	300,000
B	Group 3	Group 4	Group 5	3.9	5.5	4.2	1.6	2.5	1.9	250,000
C	Group 4	Group 6	-	4.5	3.5	-	1.4	2.4	-	180,000
D	Group 2	Group 3	Group 6	6.5	7.0	4.7	2.2	1.7	1.8	200,000

Group	Costs ($) 500 L	1000 L	2000 L	2500 L	3000 L
1	8300	12600	19100	21900	24400
2	9200	13900	21000	24100	26800
3	11700	17700	26800	30600	34200
4	10800	16400	24900	28400	31700
5	15000	22700	34400	39400	43900
6	15400	23300	35400	40500	45100

Discrete Sizes (All Items)
500 L
1000 L
2000 L
2500 L
3000 L

Total Available Production Time
H = 6200 h

Figure 1. Layout of the multipurpose plant with multiple production routes for example 5.

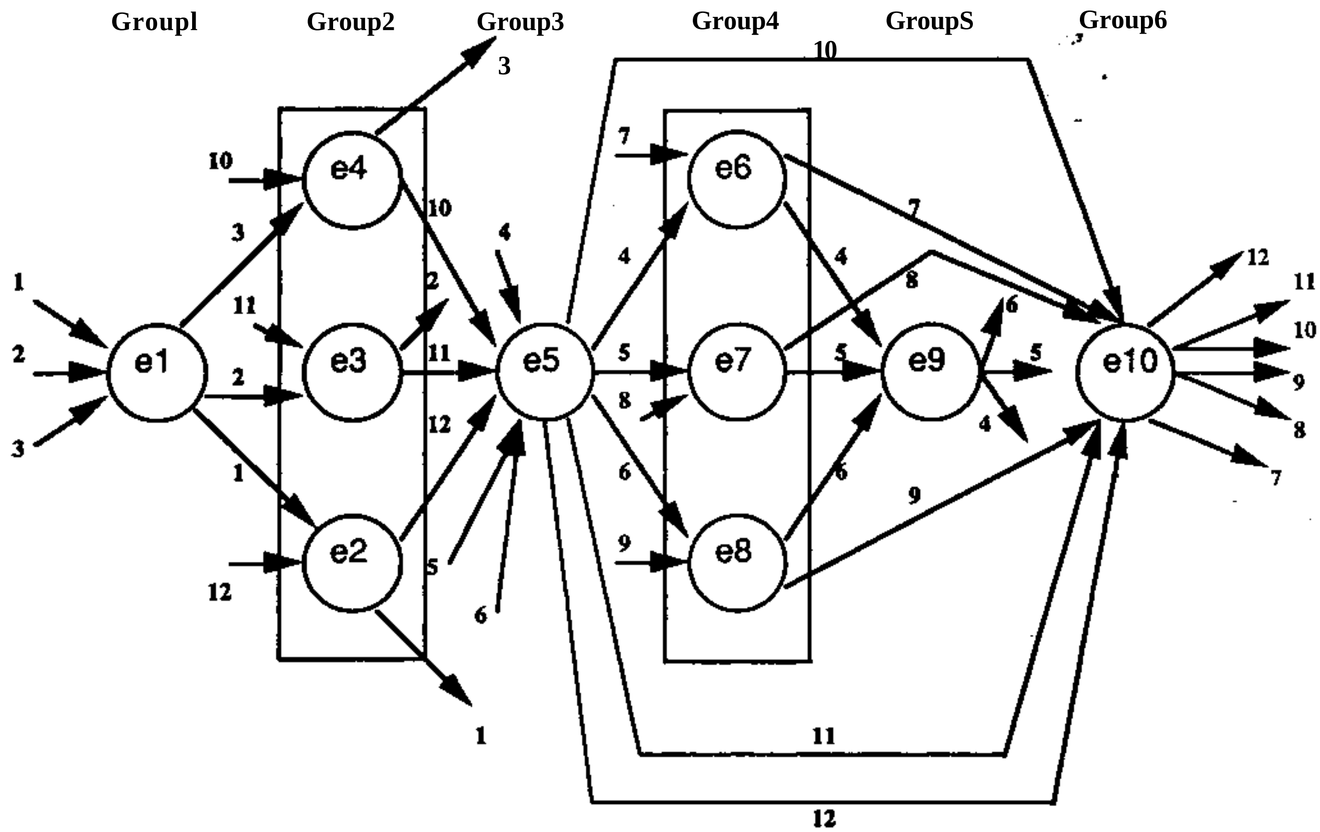

Figure A1. Layout of a multipurpose plant.

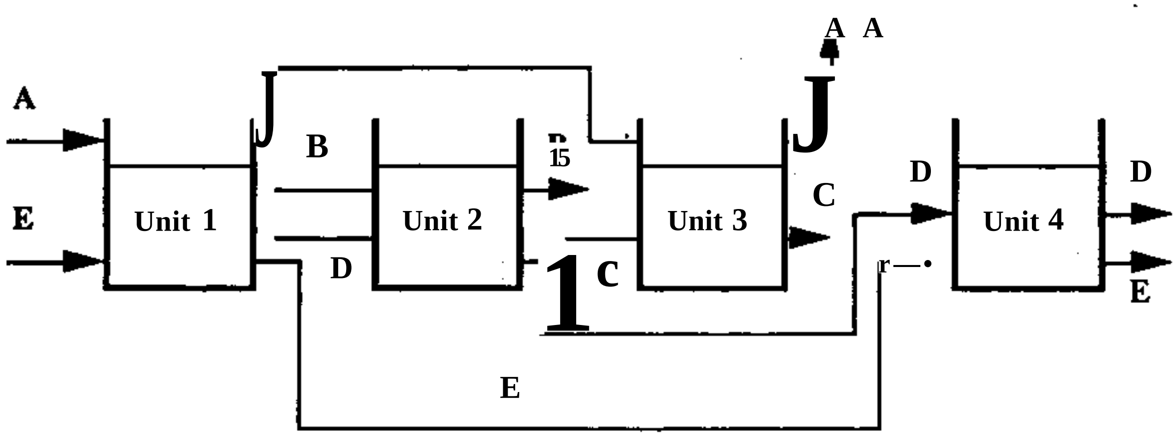

Figure A2. Graph representation of compatible products.

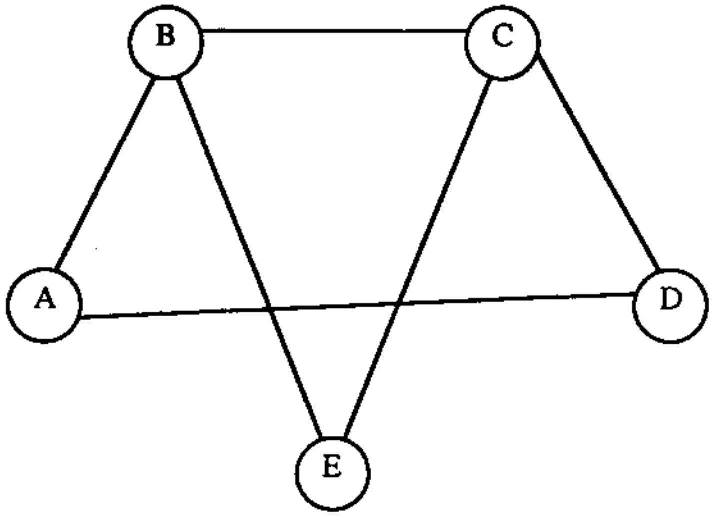

Figure A3. Maximal cliques of the graph shown in figure A2.