REFORMULATION OF MULTIPERIOD MILP MODELS FOR PLANNING AND SCHEDULING OF CHEMICAL PROCESSES

N.V. Sahinidis and I.E. Grossmann*

Department of Chemical Engineering
Carnegie Mellon University
Pittsburgh, PA 15213

June 1990

* Author to whom correspondence should be addressed.

Reformulation of Multiperiod MILP Models for Planning and Scheduling of Chemical Processes

by

N.V. Sahinidis, I. E. Grossmann

06-90-90 £, 3

ABSTRACT

A large number of planning and scheduling problems can be formulated as multiperiod MILP models which often require substantial computational expense for their sohition. This paper presents and demonstrates the value of nonstandard formulations of such problems. Based on a variable disaggregation technique which exploits lot sizing substructures, we propose a strategy for the reformulation of conventional multiperiod MILP models. The suggested formulations involve more constraints and variables but they exhibit tighter linear programming relaxations than standard approaches. The proposed reformulation strategy is applied to a model for batch scheduling and a model for long range planning. Numerical results are presented for these problems to demonstrate that - due to their tighter linear programming relaxations - the reformulations can lead to up to an order of magnitude faster computational results and make possible the solution of larger problems.

faster computational results for large problems. This happens because the resulting models exhibit tighter linear programming relaxation which results in the enumeration of a smaller number of nodes during a branch and bound search.

The paper is organized as follows. The following section provides the necessary background by describing the lot sizing problem. This not only serves as an example to illustrate the variable disaggregation ideas, but it also plays an essential role in the development of the reformulated planning and scheduling models. In Section 3, we describe the general structure of a multiperiod planning or scheduling model with fixed and variable costs and we develop the reformulation strategy based on the observation that lot sizing substructures are embedded into these models. Sections 4 and 5 present the application of the suggested technique to the scheduling and the planning problem described above. Theoretical properties of the reformulations are also given in these sections. Computational results with the new models are presented in Section 6 where the practical significance of the reformulation becomes apparent. Conclusions from this work are drawn in Section 7.

2. Reformulation and Lot Sizing

Consider a batch reactor which produces a single product with time varying demand. Set-up costs are incurred each time the reactor is utilized. Large amounts of product may be produced at early points in time in order to satisfy future demand. In this case, however, inventory holding costs have to be paid. The situation leads to a production planning problem — the lot sizing problem - where the objective is to minimize the sum of the costs of production, storage, and set-up, given that demand must be satisfied in each of NT time periods and backlogging is not allowed. For t = 1, NT, let d_t be the demand in period r, and let c_t, p_r, and hj be the set-up, unit production, and unit storage cost, respectively, in period /.

A common formulation for this problem is obtained (see Nemhauser and Wolsey, 1988) by defining x_t and s_t as the production and storage amounts in period / and by defining a binary variable y_t, indicating whether $x_t > 0$ or not This leads to the model:

Model LS:

ntin 2rf2 , ((PPr^{x}x r/ $^{+}$+ ^{n}h r ^{s}s/ $^{+}$+ ^{c}t f yYt) " (2.1)
$t = 1$

st.

$s_{r-}! + x_r = d_r + s_r$ r=l,NT   (2.2)

$x_f \leq co\ y_r$ r=l,NT   (2.3)

$s_0 = O$ (2.4)

$s_r,\ x_r \geq 0,\quad y_r e\{0,1\}$ r=l,NT   (2.5)

where co = ^V_i d$_r$ (v^NT, $t=1$) is an upper bound on x_t for all t.

Theorem 1 (Wagner and Whitin, 1958). For the lot sizing problem, there always exists a minimal cost policy with the property that x^ has one of the following values:

$$0, \quad d_r, \quad d_t + d_{t+1}, \quad d_t + d_{t+1} + d_{t+2}, \quad \ldots, \quad \sum_{T=t}^{NT} d_T.$$

Based on this result, Wagner and Whitin (1958) developed an efficient dynamic programming algorithm to search over the above discrete set of solutions to find the optimum solution of the lot sizing problem. Another alternative is to directly solve the integer program (LS). In order to efficiently solve this problem, Krarup and Bilde (1977) presented the formulation which we describe next

By defining $q/_T$ as the quantity produced in period t to satisfy the demand in period $x \geq t$, we have:

$$x_t = \sum^{NT} q_{rT} \qquad r=l,NT \qquad (2.6)$$

Problem (LS) can then be reformulated in terms of $q_{t\tau}$ and y_t as follows:

Model RLS1:

$$\min \quad \sum_{t=1}^{NT} \sum_{\tau=t}^{NT} (p_t + h_t + h_{t+1} + \ldots + h_{\tau-1})\, q_{t\tau} + \sum_{t=1}^{NT} c_t y_t \tag{2.7}$$

st.

$$\sum_{\tau=1}^{t} q_{\tau t} = d_t \qquad t = 1, NT \tag{2.8}$$

$$q_{t\tau} \leq d_\tau y_t \qquad t = 1, NT \qquad \tau = t, NT \tag{2.9}$$

$$q \in \Re_+^{NT(NT+1)/2}, \quad y \in \{0, 1\}^{NT} \tag{2.10}$$

As mentioned, the variables $q_{t\tau}$ introduced in this reformulation of model (LS) can be seen as amounts produced in period t in order to satisfy demand *for* period $\tau \geq t$. This is depicted in Fig. 1, where we show the problem representation (a) before, and (b) after the reformulation. It is clear that in (a) we have a fixed charge network. Therefore, the reformulation in (b) can be derived from the suggestions of Rardin and Choe (1979) for obtaining tighter relaxations of network flow problems with fixed charges: each variable x_t of the original formulation is now disaggregated into NT–t+1 new variables $q_{t\tau}$ ($\tau = t, NT$). The variable disaggregation in this case gives not just a tighter formulation but the absolute tightest one:

Theorem 2 (see Nemhauser and Wolsey, 1988). The solution to the linear programming relaxation of (RLS1) yields 0-1 values for the y-variables. In addition, the image in the (x, s, y)-space under the transformation (2.6) of all the points (q, y) feasible in the linear programming relaxation of model (RLS1) produces the convex hull of model (LS).

It follows from this theorem that, one only needs to solve (RLS1) as a linear program where the y-variables are relaxed to take values in the interval [0, 1] and obtain the solution to

the integer program (LS). It is interesting to note that model (RLS1) is not the only possible formulation exhibiting this property. Based on the work of Barany, Van Roy and Wolsey (1984), Martin (1987b) used separation algorithms and derived for the lot sizing problem another alternative formulation for which Theorem 2 holds. In this reformulation, the disaggregated variable X_{tr} represents the amount produced in period *t* in order to satisfy demand *up to* period *t>t.* Martin's reformulation is the following:

~~Model RLS2:~~

$$\min \sum_{r=l}^{NT} 2^* \ (Vt\ ^{x}/ + \hat{h}/\ ^{s}/ + {}^{c}t\ Yt) \qquad (2.1)$$

st.

$^{s}r\text{-}l + {}^{x}/ = {}^{d}f + {}^{s}r$	$f=l,NT$		(2.2)
$x_t \leq C_{t,NT}\ y_t$	$t = 1, NT$		(2.11)
$x_t \geq \hat{A}\text{-}_{tT}$	$/=1,NT$	$t=t, NT$	(2.12)
$\lambda_{t\tau} \leq C_{fT}\ y_r$	$f=l,NT$	$T=r,NT$	(2.13)
$\sum_{T=l}^{t} X_w \geq C_{lr}$	$/= 1,NT$		(2.14)
$SO = {}^{\circ}$			(2.4)
$s_r,\ x_r \geq 0, \quad y, € \{0,1\}$	$r=l,NT$		(2.5)
$\wedge rr \wedge O$	$r=l,NT$	$r=r,NT$ "	(2.15)

where $C_n = \wedge_{l=t}^{\tau} d_T$ are upper bounds for the disaggregated production variables X_{tr}.

In the above formulation - in contrast to the reformulation of Krarup and Bilde - the original variables x^ and s_t are not eliminated; instead the variables x^ are related to the disaggregated variables ^through the inequalities (2.12).

In addition to models (RLS1) and (RLS2), based on the work of Barany, Van Roy and Wolsey (1984), Pochet and Wolsey (1988) used the theory of strong cutting planes to derive yet another formulation for which Theorem 2 is valid. These three, slightly different representations, differ in the number of constraints and variables they include, and therefore in their computational efficiency. Of course, efficient dynamic programming techniques are available to solve the lot sizing problem (Wagner and Whitin, 1958; Zangwill, 1969). However, the above reformulations are very important when the lot sizing problem is part of a more complex planning model. The importance of reformulations (RLS1) and (RLS2) will be shown in the development of Models (Rl) and (R2) of this paper. This development is based on the observations described in the next section.

3. Strategy for the Reformulation of Multiperiod MILP Models for Planning and Scheduling

The following is a general multiperiod MILP model:

Model P:

$$\min \quad \sum_t (\alpha_t^T X_t + \$Y_t + yj\ Z_r + 5?\ V_r) \tag{3.1}$$

s.t.

Ar **Xr + B, Y, + Cr** ^{z}**r + D_t** Vr **<a,** Vr **(3.2)**

^{E}r ^{X}r + F_{t-1} Z_{t-1} + Gr Z_r+ ^{H}r Vr^b, Vr **(3.3)**

X_t ≤ AY, Vr **(3.4)**

X_t, Z_t, V_t ≥ 0, Y/^= 0 or 1 Vr **(3.5)**

where A_p B_p C_r, D_r, E_r, ¥$_p$ G_r, H_r, and a_p p_r, y_p 5_p a_r and b_r are matrices and vectors of conformable dimensions, and A is a diagonal matrix of upper bounds. The vector-variables $X_($

and V_t represent activities for each time period f, with the former being activated by the vector Y_t of 0-1 variables. The vector variables Z_t represent coupling variables between successive time periods.

Assume that the set of constraints (3.3) is of the form, or can be recast as:

$$I, = W + {}^{x}t - {}^{d}r \qquad \text{v ,} \tag{3.6}$$

$$I, = M, Z, + N, V, + O, \qquad Vr \tag{3.7}$$

$$d, = f,(X,Z,V) \qquad Vr \tag{3.8}$$

where M_r, N_t are matrices, O^ are vectors and f_t are possibly nonlinear functions. In the lot sizing terminology, equation (3.6) is a mass balance for the inventory (1^) in time period r; equation (3.7) can be used to express storage constraints; and equation (3.8) defines the demand (dp.

Observe the similarity of (3.6) and (3.4) to (2.2) and (2.3) respectively. Also observe the similarity of the objective functions of problems (LS) and (P). Obviously, lot sizing substructures are embedded in the multiperiod MILP model (P). This suggests the following **strategy for the reformulation of multiperiod MILP models:**

Step 1: Identify the presence of constraints similar to (3.6). If necessary, recast the given problem into that form by constraint manipulations.

Step 2: Disaggregate the variables X_t by introducing new variables Q_{tt} (r $\geq$ r).

Step 3: Reformulate the problem in terms of the new variables Q_{tr} and the corresponding lot sizing constraints.

The form of the constraints to be introduced in Step 3 depends on which one of the different reformulations of the lot sizing problem we choose to use in the second step of the reformulation strategy. Use of the Krarup-Bilde reformulation will introduce the following constraints which are similar to (2.6), (2.8) and (2.9):

$$X, = \sum_{x \geq t} X \; 9/T \qquad \text{v ,} \tag{3.9}$$

$$\sum_{\tau \leq t} \theta_{\tau t} = d_t \qquad \forall t \qquad (3.10)$$

$$\theta_{t\tau} \leq d_\tau \, Y_t \qquad \forall t \qquad \forall \tau \geq t \qquad (3.11)$$

If one uses Martin's reformulation for the lot sizing problem, the new constraints should have the following form (similar to (2.12) to (2.14)):

$$X_t \geq \theta_{t\tau} \qquad \forall t \qquad \forall \tau \geq t \qquad (3.12)$$

$$\theta_{t\tau} \leq C_{t\tau} \, Y_t \qquad \forall t \qquad \forall \tau \geq t \qquad (3.13)$$

$$\sum_{\tau \leq t} \theta_{\tau t} \geq C_{1t} \qquad \forall t \qquad (3.14)$$

At a first look, it may seem advantageous to use the Krarup-Bilde reformulation since it involves fewer constraints. However, if f_t in (3.8) are not constants, the demands d_t will have to be treated as variables in the new model. In that case, (3.11) is a nonlinear constraint and the reformulation would introduce nonconvexities. This difficulty can be overcome as follows:

Case I: If the functions f_t are linear, an overestimation of the d-variables can be used in (3.11) with the Krarup-Bilde reformulation.

Case II: If the functions f_t are nonlinear (possibly the result of recasting the problem as a lot sizing problem), then constraints (3.8) can be ignored by using valid upper bounds in the place of the d_t's in Martin's reformulation. The Krarup-Bilde reformulation cannot be used in this case since it would yield erroneous results due to the presence of (3.10).

In either of the above cases, the reformulation leads to the following multiperiod MILP model:

<u>Model R:</u>

$$\min \quad \sum_{t} \left(\alpha_t^T X_t + \beta_t^T Y_t + \gamma_t^T Z_t + \delta_t^T V_t \right) \qquad (3.1)$$

s.t.

$$A_t\ X,\ +\ B,\ Y,\ +\ C_t\ Z_t\ +\ D,\ V,\ \leq\ a, \qquad \text{Vr} \qquad (3.2)$$

$$X_t\ \leq\ A\ Y, \qquad \text{Vr} \qquad (3.4)$$

$$I_r\ =\ I_{t}.j\ +\ X_r\ -\ d, \qquad \text{Vr} \qquad " \ (3\text{-}6)$$

$$I,\ =\ M,\ Z,\ +\ N,\ V,\ +\ O, \qquad \text{Vr} \qquad (3.7)$$

$$g_t(X_t, \theta>_{rT}, d_r)\ \leq\ 0 \qquad \text{Vr} \qquad \text{Vr}\geq r \qquad (3.15)$$

$$X_r,\ Z,,\ V,\ \geq\ 0, \quad Y_t\ =\ 0\ \text{or}\ 1 \qquad \text{Vr} \qquad (3.5)$$

$$I,\ \geq\ 0 \qquad (3.8)$$

$$Q_n\ \geq\ 0 \qquad \text{Vr} \qquad \text{Vr}\geq r \qquad (3.16)$$

$$d_r\ =\ f_r(X, Z, V) \qquad \text{Vr} \qquad (3.8)$$

where g_t is a linear constraint set which denotes either the Krarup-Bilde constraints in (3.9) - (3.11), or Martin's constraints in (3.12) to (3.14), depending on which of the available reformulations of the lot sizing part of the problem is used. No matter what the form of g_t is, the following theorem can be established for the tightness of the LP relaxation of model (R):

Theorem 3. The optimal cost of the linear programming relaxation of model (R) is not lower than the optimal cost of the linear programming relaxation of model (P).

Proof: Consider any point (X, Y, V, Z, 9) which is feasible in the linear programming relaxation of model (R). It follows that the point (X, Y, V, Z) satisfies the constraints (3.2) to (3.5) which define the feasible region of the linear programming relaxation of model (P). In that case, the feasible space for the linear programming relaxation of model (R) is contained within the feasible space of the linear programming relaxation of model (P) and the theorem holds. •

4. Multiperiod MILP Models for Scheduling Process Operations

Consider a general batch processing system where a set of products is to be produced from a set of feedstocks according to a prespecified sequence of elementary operations (tasks). The problem has been addressed by Kondili *et al.* (1990) on the basis of a state-task network

(STN) representation. Fig. 2 represents a conventional process flowsheet. The corresponding STN, as given by the above authors, is shown in Fig. 3. An STN has two types of nodes; namely, the state nodes (*s* = 1, NS), representing the feeds, intermediate and final products, and the task (operation) nodes (/ = 1, NO), representing the operations. Each task is described by a recipe: type and percentage of input and output states, and duration of the processing. Finally, each of a number of units (/ = 1, NU) is able to perform a number of operations (że L_j).

Given are the costs for purchasing feedstocks, processing intermediates, storing material, and the prices of the products. Also given are bounds for the availabilities of the raw materials and demands of products for each time period. Constraints on the availability of intermediate storage may also be specified. The goal is to optimize a given economic objective function over a short range horizon consisting of NT time periods of the same duration h. This requires to determine the following items:

(i) the timing of the operations for each unit (i.e. which task, if any, each unit performs at each time),

(ii) the flow of material through the network (purchases, intermediate storage, sales).

The following notation will be used to describe the model:

Parameters:

a,Yf — is the variable part of the production cost of operation *i* in unity during time period r,

PJYJ — is the fixed part of the production cost (set-up cost) of operation / in unit *j* during time period r,

y_{st} — is the storage cost for the product in state *s* and time period r,

b_{st} — is the purchase price for the product in state *s* and time period r,

X_{st} — is the sales price for the product in state *s* and time period r,

pjTy — the proportion of input of task *i* from state *s* when task *i* is executed in unit /;

$\overline{P}ijs$ — ^e proportion of output of task *i* in state *s* when task *i* is executed in unit *j*\

ICy_r — maximum storage capacity for state *s* during time period r,

IT_j — set of tasks which can be performed by unity;

JU; — set of units which can process task i;

Pjy — is the duration of operation *i* in unity;

PL_{st}	lower bound for the purchase of raw material in state *s* at the beginning of time period r,
PU_{st}	upper bound for the purchase of raw material in state *s* at the beginning of time period r,
S_{st}	demand for product in state *s* at the beginning of time period r,
T_s	set of tasks (operations) receiving material from state *s;*
$\bar{T}_s$	set of tasks producing material in state s;
V_{ij}	capacity of unit *j* when performing task i.

Variables:

I_{st}	amount of raw material in storage in state *s* during time period r,
P_{st}	amount of raw material in state *s* which is purchased at the beginning of time period r,
W_{ijt}	is the amount of material which starts undergoing task *i* in unit *j* at the beginning of time period r,
Y_{ijt}	is 1 if unit *j* starts processing task *i* at the beginning of time period r, and 0 else.

When the demands for the products are given, the following MILP model can be used to describe the problem:

<u>**Model PI:**</u>

$$\min \sum_{i=1}^{NO} \sum_{j=1}^{NU} \sum_{t=1}^{NT} (\ldots \, ijt \; ** \, ijt \; \bullet \; r_{ijt} \; * \; \ldots t) \qquad (4.1)$$

$$+ \sum_{s=1}^{NS} \sum_{t=1}^{NT} (1st\ hi + \$st\ Pst - 1st\ S_{st})$$

s.t.

$$\sum_{i \in I_j} Y_{ijt} \le 1 \qquad \forall j \qquad \forall t \qquad (4.2)$$

$$Y_{ijt} \le 1 - Y_{i'jt'} \qquad \forall j \qquad \forall i' \in I_j \qquad t' = t-1, \ldots, t-p_{ij}+1 \qquad (4.3)$$

$$0 \leq W_{ijt} \leq V_{ij}\, Y_{ijt} \qquad \forall i \qquad \forall j \qquad \forall t \tag{4.4}$$

$$0 \leq I_{st} \leq IC_{st} \qquad \forall s \qquad \forall t \tag{4.5}$$

$$PL_{st} \leq P_{st} \leq PU_{st} \qquad \forall s \qquad \forall t \tag{4.6}$$

$$\begin{aligned} I_{st} = I_{s,t\text{-}1} &+ P_{st} - S_{st} \\ &+ \sum_{i \in \overline{T}_s} \sum_{j \in JU_i} \bar{\rho}_{ijs}\, W_{ij,t\text{-}p_{ijs}} \\ &- \sum_{i \in T_s} \sum_{j \in JU_i} \rho_{ijs}\, W_{ijt} \qquad \forall s \qquad \forall t \end{aligned} \tag{4.7}$$

$$W_{ijt},\, I_{st},\, P_{st} \geq 0, \qquad Y_{ijt} = 0 \text{ or } 1 \tag{4.8}$$

The objective in the above model is to minimize the total cost which consists of four terms: variable and fixed production cost, inventory cost, and the cost for purchasing raw materials. The last term in (4.1) denotes the sales revenue which is a constant since the demands S_{st} are given. Equation (4.2) enforces the condition that at most one operation may be started at any unit in the beginning of a time period. According to (4.3) no preemption is allowed: once operation i' begins, it may not be interrupted in order to execute any other operation i. The variable upper bounds (4.4) are used to ensure that an operation may start only when the corresponding binary variable is assigned a value of one. Zero, finite or unlimited intermediate storage conditions are imposed through (4.5) while constraints (4.6) express lower and upper bounds on the availability of the raw materials. Finally, (4.7) is a mass balance equation between time periods for each state.

Kondili *et al.* (1990) address a slightly more general problem with the demands being variables. We have assumed that there are prespecified, time varying demands. This is indeed the case for a scheduling problem with a short time horizon; the plant has to produce material according to the decisions of a higher level planning model. Simultaneous planning and scheduling here would require looking at a long range horizon and therefore introducing a prohibitively large number of time periods. Also, our model differs to the model of Kondili *et al.* in the way the logical constraints are imposed in (4.2) and (4.3). An advantage of this formulation is that by using (4.2) the special structure of special ordered sets of type 1 can be exploited (see Beale, 1979).

For illustration purposes, consider the small example of the batch process described by the state task network of Fig. 4. There is one feed, one intermediate and two final products which are involved in three processing tasks (all mass balance coefficients $Pij_s = \overline{py}_S = 1$). There are three available units and each one is suitable for a different task. Demand is specified over a short range horizon consisting of 12 time units. The problem data are given in Table 1. The MILP corresponding to this problem involves 36 binary variables, 97 continuous variables and 90 constraints. The solution was obtained in 8.6 seconds on an IBM-3090 by solving model (PI) using MPSX-MIP/370 (IBM, 1988). The optimal schedule with a profit of 3,230 is shown in Fig. 5. The number above each line segment identifies the processing task, whereas the number below it is the amount of material which undergoes the corresponding task.

In order to expedite the solution of large problems, Kondili *etal.* (1990) developed dominance criteria which reduce the number of nodes to be examined during a branch and bound enumeration procedure. Here, we present a non-standard formulation in order to tighten the linear programming relaxation bounds.

Observation

In equation (4.7), the term S_{st} for the sales is nonzero when (4.7) is applied to those states which correspond to final products only. Then, for any final product s, (4.7) is:

$$ht = W\text{-}l \text{ " } {}^{s}{*}r + \sum_{i \in \overline{T}_s} \sum_{j \in JU_i} \overline{P}ijs \; {}^{w}//{>}p//_5 \qquad Vr \qquad (4.9).$$

There is some similarity between (4.9) to (3.6); the only difference is the presence of more than one production terms in (4.9). Furthermore, (4.9) can be equivalently rewritten as:

$$Mjst = hjsM - {}^{s}ijst + {}^{w}//{>}p_{/y}, \qquad V, \qquad /e\overline{T}_5 \qquad /eJU/ \qquad (4.10a)$$

$$^{s}st = \overline{\sum_{ie\overline{T}_s}} \; \overline{\sum_{jeJJi}} \; \overline{P}ijs \; {}^{s}ijst \qquad Vr \qquad (4.10b).$$

Now, (4.10a) is identical to (3.6) which suggests that we should disaggregate each one of the production terms appearing in (4.9). Let us then define ^*hx* ^ the amount which starts undergoing operation j in unity at the beginning of time period r in order to satisfy demand for a subsequent period *x≥t*. The new variables must satisfy constraints analogous to (3.9) to (3.11):

$$W_{ijt} = \sum_{r>r} \text{«}_{it\tau} \qquad \text{VieI*} \qquad \text{V/} \qquad \text{Vr} \qquad (4.11)$$

$$\text{II} \quad \sum_{I} \quad \sum_{j} \quad \sum_{V\wedge t-Pijs} \bar{P}_{ijs} \; {}^{f}\text{®}iJTt = \$St \qquad \forall s \in S^{*} \qquad \text{Vr} \qquad (4.12)$$

$$\bar{P}_{ijs} \; toijn \; \wedge \; {}^{rr}\wedge^{n} \{ \; SsT> \; {}^{v}\dot{y} \; 1 \; {}^{Y}ijt \qquad \text{V}\wedge\text{eS*} \qquad \text{V/,;, } t \qquad \text{V T} \geq \text{r+p}\wedge \qquad (4.13)$$

where S* is the subset of states in the network corresponding to final products and I* is the set of operations producing final products. Then the model after the disaggregation of variables becomes:

<u>Model Rl:</u>

$$\min \sum_{i=1}^{NO} \sum_{j=1}^{NU} \sum_{t=1}^{NT} (MjtVfijt + \qquad tojtYijt) \qquad (4.1)$$

$$+ \sum^{NS} \sum^{NT} (\qquad + \delta_{st} \, P_{st} - \lambda_{st} \, S_{st}) $$

s.t.

$$\sum_{i \in I_j} {}^{Y}ijt \; * \; {}^{l} \qquad \text{V}\dot{y} \qquad \text{Vr} \qquad (4.2)$$

$$Y_{ijt} \leq 1 - Y\wedge' \qquad \text{V/} \qquad \text{VZ/elTy} \qquad t' = \text{M}, ..., \text{f-p/y+1} \qquad (4.3)$$

$$0 \leq W,y_{f} \leq V,y \; Y_{,yr} \qquad \text{Vi} \qquad \text{V/} \qquad \text{Vr} \qquad (4.4)$$

$$0 \leq I\wedge_{r} \leq ICy, \qquad \text{V5} \qquad \text{Vr} \qquad (4.5)$$

$$PL_{5f} \leq P_{st} \leq PU_{,r} \qquad \text{V5} \qquad \text{Vr} \qquad (4.6)$$

$$+ \underset{ie\overline{T}_s}{1} \quad \underset{je\,\mathrm{JU}_i}{X} \quad \overline{Pijs} \quad {}^{\mathrm{TM}}ij,t\text{-}p_{ijs}$$

$$- \sum_{i\in \mathrm{T}_s} \quad \underset{j\in \mathrm{JU}_i}{X} \quad Pijs \quad {}^{\mathrm{W}}(/\mathrm{Y} \qquad \mathrm{V5} \qquad \mathrm{Vr} \qquad (4.7)$$

$$\mathrm{W}_{ijt} = \sum \omega_{ijt\tau} \qquad \mathrm{V/eF} \qquad \mathrm{Vy} \qquad \mathrm{Vr} \qquad (4.11)$$

$$\sum_i \sum_j \sum_{\tau \leq t - p_{ijs}} \overline{\rho}_{ijs}\, \omega_{ij\tau\tau} = S_{st} \qquad \mathrm{V^{\wedge}eS^*} \qquad \mathrm{Vr} \qquad (4.12)$$

$$\overline{Pijs} \ <^*ijtT \ * \ \mathrm{mi}"\ (\ ^{\mathrm{S}}{*}\mathrm{T}>\ ^{\mathrm{V}}{\cdot}\mathrm{y}\ )\ \% \qquad \mathrm{VseS^*} \qquad \mathrm{V/,y,\ f} \qquad \mathrm{VT{\geq}r+p^{\wedge}} \qquad (4.13)$$

$$\mathrm{W_{jy}},\ \mathrm{I}_{5/}, \mathrm{P}_5,\ \geq 0, \qquad \mathrm{Y_{jyr}} = \mathrm{Oor}\ 1. \qquad (4.8)$$

$$\omega_{ijt\tau} \geq 0 \qquad (4.14)$$

In the above model, one can use (4.11) to eliminate some of the variables and constraints of the problem. Even when this is done, the new model contains more variables and constraints. However, this increase in variables and constraints is polynomial in the number of final products, tasks, and time periods, while at the same time the new model satisfies the following theorem:

<u>Theorem 4.</u> The optimal cost of the linear programming relaxation of model (Rl) is not lower than the optimal cost of the linear programming relaxation of model (PI), and it may be strictly larger.

The first part of the theorem follows as a consequence of Theorem 3, while the second part will be proved in the section describing the computational results which indeed indicate that the new relaxation is tighter for all the examples solved. Notice that there is no guarantee that the reformulation will always yield a tighter linear programming relaxation. In fact, for the special case where the demand for the products is specified at the final time period t = NT *{i.e.* $S_{st} = 0$ *Vse* S* and for t = 1, 2,..., NT-1), the disaggregated variables $co/_{j\mathrm{rT}}$ take a value of zero (from (4.13)) for $\tau = 1, 2, ..., NT\text{-}1$, and $t \leq \tau$. In this case, model (R1) reduces to the original model (P1) and the reformulation has no effect.

5. Multiperiod MILP Models for Long Range Planning

A network consisting of a set of NP chemical processes which can be interconnected in a finite number of ways is assumed to be given. The network also involves a set of NC chemicals which include raw materials, intermediates and products. The processes will be interconnected by a total of NS streams to represent the different alternatives which are possible for the processing and the purchases and sales from NM different markets. It will be assumed that the material balances in each process can be expressed linearly in terms of the production rate of a main product, which in turn defines the capacity of the plant.

The objective function to be maximized is the net present value of the project over a long range horizon consisting of a finite number of NT time periods during which prices and demands of chemicals, and investment and operating costs of the processes can vary. The operating cost of a plant will be assumed to be proportional to the flow of its main product. As for the investment costs of the processes and their expansions, it will be considered that they can be expressed linearly in terms of the capacities with a fixed charge cost to account for the economies of scale.

In the description of the model, the following notation will be used:

Indices:

i	process (i = 1, NP);
t	time period (t = 1, NT);
j	chemical (j = 1, NC);
k	stream in the network (k = 1, NS);
l	market (l = 1, NM).

Parameters:

NP	number of processes in the network;
NT	number of time periods considered;
NM	number of markets;
NC	number of chemicals in the network;

NS number of streams in the network;

I(j) the index set of streams of chemical *j* which are produced in the complex;

O(j) the index set of streams of chemical *j* which are consumed in the complex;

L} the index set of streams corresponding to inputs and outputs of process /;

m i stream corresponding to the main product of process *i (m i e L* $_z$-);

Q/Q existing capacity of process *i* at time *t* = 0;

QE^ lower bounds for the capacity expansions;

QE_{it}^{U} upper bounds for the capacity expansions;

μ_{ik} material balance coefficients characteristic of each process / and stream /r,

(XJY variable term of investment cost [$ / unit of capacity installed];

PJY fixed term for the investment cost [$];

$d_{m_i t}$ unit operating cost[$ / unit of production amount of the main product];

/
Jj$_t$ prices of sales of the chemical *j* in market / during time period *t* [$ / unit sold];

/
r^ prices of purchases of the chemical *j* in market / during time period *t* [$ / unit purchased];

NEXP(*z*) the maximum allowable number of expansions for process i;

CI(r) the capacity investment limitation corresponding to period *L*

Variables:

y_{it} decision variable which is 1 whenever there is an expansion for process / at the beginning of time period r, and 0 otherwise;

Q_{it} total capacity of the plant of process / which is available in period r,

QE_{it} capacity expansion of the plant of process / which is installed in period r,

P_{jt}^{l} amount of producty purchased from market / at the beginning of period r;

S_{jt}^{l} amount of product *j* sold to market / at the beginning of period r,

W^$_r$ amount of flow of stream *k* during time period *t.*

A multiperiod MILP model for the long range planning problem is as follows Sahinidis *et al.* (1989):

<u>Modcl P2:</u>

max NPV = - 2, 2, «*&QEfc + Pay*) - L LSm$_{it}$Wm$_{it}$ _ (5.1)
i = 1*l*=\ ˙1 = 1 /=1

$$+\sum^{NM}\sum^{NC}\sum^{NT}(\gamma^{l}_{jt}S^{l}_{jt}-\Gamma^{l}_{jt}P^{l}_{jt})$$
/ = 1 7˙= 1 ' = 1

S.t.

y$_{l7}$ Qpf, ≤ QEfc ≤ QE,? y/, ˙ = LNP ' = LNT (5.2)

Qif = Qi,r-/ $^{+}$ Q^{E}/r '˙= LNP f = 1.NT (5.3)

$Q_{it} \geq W_{m_i t}$ ż=l,NP r=l,NT (5.4)

$W_{kt} = \mu_{ik} W_{m_i t}$ $k\in L$fMm,-} i = 1.NP r = 1.NT (5.5)

NM *t* NM *l*
X P+ Z W$_{kt}$ = S S$_{yt}$ + Σ W$_{fa}$ y=l,NC r=l,NT (5.6)
1=1 kel(j) /=i *keOU)*

a$_y$r - ^ - V ; = 1 N C r = i f N T *l* = 1,NM (5.7)
d

NT
Σ *Yit* * NEXP(/) i˙€ r C {1,2, ... NP} (5.8)
t = 1

NP
2 (<x$_{l7}$ QB$_{l7}$+P$_{ft}$ y//) ^ CI(r) re Tc {1,2, ... NT} (5.9)
i = 1

y_{it} = 0 or 1 i˙ = 1,NP r = 1,NT (5.10)

Q$_{it}$, QE$_{it}$, W$_{kt}$, Py(, Sj, ≥ 0 (5 1 D

In equation (5.1), the net present value is defined as the sum of the investment cost, the operating cost, the sales revenue and the cost for purchasing the raw materials. All the coefficients are discounted at a specified interest rate and include the effect of taxes in the net

present value. Constraint (5.2) is a variable lower and upper bounding constraint for the capacity expansions. A zero-value of the binary variables y^ forces the capacity expansion of process *i* at period *t* to zero, *i.e.* QE/$_r$ = 0. If the binary variable is equal to one$_r$ a capacity expansion between the specified bounds is performed. Equation (5.3) simply defines the total capacity, Q$_{ir}$, which is available for process *i* at each time period *t*, while Q/Q *^{S} the initial capacity (zero for nonexisting processes). Constraint (5.4) expresses the condition that the operating level of a process - expressed in terms of the flow of its main product - cannot exceed the installed capacity. The material balances in each plant are given by the linear relations (5.5): the flow of each product is proportional to the flow of the main product of the process, where ji^ are positive constants characteristic of each process. The material balances for each chemical in the entire network are given in (5.6) according to which the total amount of a chemical's purchases from the various markets plus the amounts produced within the network must be equal to the sum of sales and the total consumption within the network. Constraints (5.7) express the lower and upper bounds for the availability of raw materials and the demand of the products. Finally, constraints (5.8) and (5.9) express limits on the number of expansions of some processes and on the capital available for investment during some time periods, respectively.

Consider, as an example, a chemical complex involving 10 processes and 6 chemicals. None of these processes is assumed to have an existing capacity. The network showing all the alternatives for this complex is shown in Fig. 6. Chemical 6 is to be produced in 4 periods, each having a length of 2 years and various constraints on the chemical demands and prices. The corresponding MILP model involves 40 binary variables, 174 continuous variables and 198 rows. The optimum configuration for an instance of this problem considered by Sahinidis *et al.* (1989) is shown in Fig. 7 and was obtained by solving model (P2) using MPSX-MIP/370 (IBM, 1988). The computational requirements were only 2 seconds on an IBM-3090.

For large process networks, however, the computational expense can be high. For example, a network with 40 processes, 50 chemicals, 2 markets and 5 time periods would involve 200 binary variables, and approximately 1000 continuous variables and 1200 constraints. Since most of the alternatives embedded in such a model are feasible, a large

number of nodes must usually be examined in a branch and bound search. Therefore, there is a clear incentive to develop efficient computational strategies since this allows the examination of a greater variety of scenarios with the planning model. Sahinidis *et al.* (1989) have compared the performance of several computational strategies including branch and bound, strong cutting planes followed by branch and bound, Benders decomposition and strong cutting planes followed by Benders decomposition. For the test problems which were considered, the combination of integer cuts, strong cutting plane generation and branch and bound was found to be the most efficient strategy for solving large-scale problems to optimality.

In order to obtain further significant reductions in the computational effort, we take a different approach in this paper by developing an alternative formulation for the problem. Notice that equation (5.3) has the form of constraint (3.3), with Q_{it} playing the role of the variables Z_t and QE_{it} taking the role of the variables X_t. Also note the analogy between equations (5.2) and (3.4). Although inventory variables are not explicitly involved as in equation (3.6), we propose to disaggregate the capacity expansion variables based on the following observations.

The Main Observations

Let us assume that, for the long range planning problem, there are zero lower bounds and infinite upper bounds for the capacity expansions (5.2), no limits on the number of expansions (5.8) and no constraints on the investment (5.9) – these assumptions will be removed later in the paper. Refer now to Fig. 6 and imagine for a moment that all flows of chemicals (W_{kt}, P^l_{jt}, S^l_{jt}) in the network have been fixed in such a way that material balances (constraints (5.5) to (5.7)) are satisfied for all time periods. Then every process can be isolated from the rest of the network and the design problem for each process i becomes: "Find the cheapest capacity expansion sequence (QE_{it}, t = 1,NT) that will allow production of the prespecified flows of chemicals (W_{kt}, P^l_{jt}, S^l_{jt})". Mathematically the problem reduces to:

Mnriel P3-i:

$$\min \sum_{t=1}^{NT} (\alpha_{it} QE_{it} + \beta_{it} y_{it}) \qquad (5.12)$$

s.t.

$$QE_{it} \le U\, y_{it} \qquad r=l,NT \qquad (5.13)$$

$$Q_{i,t-1} + QE_{it} = Q/r \qquad ' = \wedge\ m \qquad (5.14)$$

$$Q_{it} \ge W_{m_i t} \qquad r=l,NT \qquad (5.15)$$

$$QE_{/r}\ ,\ Q_{/r} \ge 0, \quad y_{j7} = 0 \text{ or } 1 \qquad t = 1.NT \qquad (5.16)$$

where U is a large positive quantity.

The objective in (5.12) is to minimize the investment cost of process / for the given flows of the main product in the right hand side of (5.15). Assume, for a moment only, that these flows are such that:

$$Q/0 * W_{m/1} \le W_{m_i 2} \le \ldots \le W_{m_i)NT} \qquad (5.17)$$

By letting:

$$SQJV = Q_{it} - W_{m/f} \qquad t= 1,NT \qquad (5.18)$$

$$d_{it} = W_{m/f} \text{ " } W_{m_i >M} \qquad r=l,NT \qquad (5.19)$$

and using the convention that $W_{mj}o = Q_{i'0}$, then $SQ/_r \ge 0$ implies (5.15) and (P3-i) can be transformed into the following equivalent lot sizing problem:

Model P4-i:

$$\min \sum_{t=1}^{NT} (\text{"//}QE/z+\wedge y//) \qquad (5\text{-}12)$$

s.t.

$$QE_{it} \le U\ y/, \qquad r=l,NT \qquad (5.13)$$

$$SQ_{i,t-1} + QE_{it} = d_{it} + SQ_{it} \qquad '= LNT \qquad (520)$$

$$SQ/O = 0 \tag{5.21}$$

$$QE_{/r}\ ,\ SQ_{/r} \geq 0, \quad y_{it} = 0 \text{ or } 1 \qquad r= 1,NT \tag{5.16}$$

In the lot sizing terminology, we can view $SQ_{,-r}$ as the "inventory" of capacity, *i.e.* excess of capacity installed at early times in order to serve demand during subsequent time periods. At the same time, the QE/f's can be regarded as "production" of capacity in order to satisfy some "demand" for capacity as determined by the flows of the main products ($W_{m}i$^) in (5.19). For example, if there is no capacity initially installed and if W_{mij} = (10, 15, 18, 20), then the demand for capacity is: $d_{z/}$ = (10,5, 3,2). In the general case - when (5.17) may not hold - this demand for capacity can be obtained as follows:

1) Subtract any existing capacity (Q/Q) fr°m W^r« If positive, let this difference be called additional required capacity, m_{7r}. Then:

$$m_{it} = \max\,(0,\ V/_{mit} - Q_{z0}) \qquad t = 1.NT \tag{5.22}$$

2) For each time period f, find the maximum additional cquired capacity during all previous time periods; this maximum is:

$$^{M}it = {}_{T}{}^{TM}{\ll}_{1}\,{}^{m}{*}T = \max\,(\ M_{z>1},\ m_{z>1}\ ) \qquad t = 1>NT \tag{5>23}$$

where m_zQ = M_ZQ = 0.

3) The demand, d_{z-r}, for capacity during time period *t* is the difference between the current additional capacity requirements (m_{z-r}) and the maximum additional capacity requirements up to the previous time period (M_z^), provided this difference is positive:

$$d_{it} = \max\,(0,\ m_{/r} - M_{it}) \qquad t = 1,NT \tag{5.24}$$

As an example, consider the case where the installed capacity is 3 units and W^^ = (10, 8, 9, 12). Then it follows from the above equations that the demand for capacity is $d_{/r}$ = (7, 0, 0, 2). The equivalence of problems (P3-i) and (P4-i) - with the demands $d_{/r}$ obtained through (5.22) to (5.24) - for values of the flows not necessarily satisfying (5.17) is established by the following theorem (the proof is given in Appendix A):

<u>Theorem 5.</u> Problems (P3-i) and (P4-i) have the same optimal solution.

Based on the above theorem, Sahinidis and Grossmann (1989) used the Krarup-Bilde reformulation (RLS1) of the lot sizing substructures of the model. However, since'in this case the demands in (5.22) to (5.24) involve nonlinear functions, this gave rise to a nonconvex NLP reformulation of model (P2). Here, we will make use of Martin's reformulation (model (RLS 2)) in order to present an MILP reformulation of the problem. As indicated in the description of problem (P4-i), the variables QE^ denote "capacity production" and therefore correspond to the production variables x_t of model (LS). Then, in order to apply the reformulation, let us disaggregate the capacity expansions by defining the variable $cp_{/t}r$ as capacity expansion of plant *i* made in period *t* in order to serve production requirements *up to* period x (x ≥ *i)*. These variables correspond to the variables $*k_{tT}$*of* model (RLS2) and therefore they have to satisfy the following constraints:

$QE_{/r} \geq 9/JX$	$i = 1.NP$	$t = 1,NT$	$x \geq t$	(5.25)
$\varphi_{it\tau} \wedge Q_{rc}\ Y_{it}$	$i \wedge 1,NP$	$t = 1,NT$	$\tau \geq t$	(5.26)

which are completely analogous to (2.12) and (2.13), respectively. Furthermore, from the definition: $Cj_r = X \overset{t}{f}\text{-}1 \wedge T^{anc} \wedge *^n$ conjunction to equation (A-8) of Appendix A, it follows that a valid relaxation of (2.14) is the following constraint:

$$\sum_{T=1}^{t} <Pm \geq {}^{W}/n_z r \text{ " } QiO \qquad / = 1.NP \qquad t = 1,NT \qquad (5.27)$$

Finally, the new variables must be nonnegative:

$$(p/rr \wedge 0 \qquad i = 1,NP \qquad t = 1,NT \qquad x \geq t \qquad (5.28)$$

By including constraints (5.25) to (5.28) in model (P2), the reformulation of the long range planning model is then the following multiperiod MILP model:

Reformulated Model R2:

$$\max \quad NPV = -\sum_{i=1}^{NP}\sum_{t=1}^{NT}(\alpha_{it}\,QE_{it} + \beta_{it}\,y_{it}) - \sum_{i=1}^{NP}\sum_{t=1}^{NT}\delta_{m_i t}\,W_{m_i t} + \sum_{l=1}^{NM}\sum_{j=1}^{NC}\sum_{t=1}^{NT}(\gamma_{jt}^{l}\,S_{jt}^{l} - \Gamma_{jt}^{l}\,P_{jt}^{l}) \qquad (5.1)$$

s.t.

$$y_{it}\,QE_{it}^{L} \le QE_{it} \le QE_{it}^{U}\,y_{it} \qquad i = 1,NP, \quad t = 1,NT \qquad (5.2)$$

$$W_{kt} = \mu_{ik}\,W_{m_i t} \qquad k \in L_i \setminus \{m_i\} \quad i = 1,NP \quad t = 1,NT \qquad (5.5)$$

$$\sum_{l=1}^{NM} P_{jt}^{l} + \sum_{k \in I(j)} W_{kt} = \sum_{l=1}^{NM} S_{jt}^{l} + \sum_{k \in O(j)} W_{kt} \qquad j = 1,NC \quad t = 1,NT \qquad (5.6)$$

$$\left.\begin{array}{l} a_{jt}^{l,L} \le P_{jt}^{l} \le a_{jt}^{l,U} \\ d_{jt}^{l,L} \le S_{jt}^{l} \le d_{jt}^{l,U} \end{array}\right\} \qquad j=1,NC \quad t = 1,NT \quad l = 1,NM \qquad (5.7)$$

$$\sum_{t=1}^{NT} y_{it} \le NEXP(i) \qquad i \in I' \subseteq \{1,2, \ldots NP\} \qquad (5.8)$$

$$\sum_{i=1}^{NP} (\alpha_{it}\,QE_{it} + \beta_{it}\,y_{it}) \le CI(t) \qquad t \in T' \subseteq \{1,2, \ldots NT\} \qquad (5.9)$$

$$QE_{it} \ge \varphi_{it\tau} \qquad i = 1,NP \quad t = 1,NT \quad \tau \ge t \qquad (5.25)$$

$$\varphi_{it\tau} \le C_{it\tau}\,y_{it} \qquad i = 1,NP \quad t = 1,NT \quad \tau \ge t \qquad (5.26)$$

$$\sum_{\tau=1}^{t} \varphi_{i\tau t} \ge W_{m_i t} - Q_{i0} \qquad i = 1,NP \quad t = 1,NT \qquad (5.27)$$

$$y_{it} = 0 \text{ or } 1 \qquad i = 1,NP \quad t = 1,NT \qquad (5.10)$$

$$Q_{it},\ QE_{it},\ W_{kt},\ P_{jt}^{l},\ S_{jt}^{l} \ge 0 \qquad (5.11)$$

$$\varphi_{it\tau} \ge 0 \qquad i = 1,NP \quad t = 1,NT \quad \tau \ge t \qquad (5.28)$$

The model contains the definition of the net present value (equation (5.1)), the variable lower and upper bounds on the capacity expansions (constraints (5.2)) and the material balances (constraints (5.5) to (5.7)). The constraints on the number of expansions (5.8) and the budget constraints (5.9) are also included. Constraint (5.25) expresses the obvious fact that the capacity expansion QE_{it}^{T} in period t to satisfy demand up to period T cannot exceed the capacity expansion QE_{it} during period t. Constraint (5.27) is now used instead of constraint (5.4) and it implies that capacity cannot be devoted to production during time period t unless it was previously acquired for this purpose.

The upper bounds C_{it}^{U} for the capacity expansions in (5.26) must be postulated *a priori* and they are not known. However, valid upper bounds for the capacity expansions can be evaluated by maximizing the individual production rate of each process i (i = 1, NP) for each time period t (t = 1, NT) by solving the following linear program:

$$\omega_{it} = \max W_{m_i t} \tag{5.29}$$

s.t.

$$W_{kt} = \mu_{ik} W_{m_i t} \qquad k \in L_i \setminus \{m_i\} \tag{5.5}$$

$$\sum_{l=1}^{NM} P_{jt}^{l} + \sum_{k \in I(j)} W_{kt} = \sum_{l=1}^{NM} S_{jt}^{l} + \sum_{k \in O(j)} W_{kt} \qquad j = 1, NC \tag{5.6}$$

$$a_{jt}^{lL} \le P_{jt}^{l} \le a_{jt}^{lU}, \quad d_{jt}^{lL} \le S_{jt}^{l} \le d_{jt}^{lU} \qquad j = 1, NC \quad l = 1, NM \tag{5.7}$$

$$W_{kt},\ P_{jt}^{l},\ S_{jt}^{l} \ge 0 \tag{5.11}$$

In this LP model the flow of the main product of a process is maximized subject to mass balances around the entire network. If finite bounds are specified for the inequalities (5.7), the solution will always be bounded. In addition, this LP has special structure. It is a *processing network* for which special solution algorithms are available (Koene, 1983; McBride, 1985; Chen and Enguist, 1988).

Then the upper bounds for the capacity expansions are:

$$C_{in} = \max \{ 0, \min \{ QE^\wedge, \max_{\bar{\lambda} = \tau,\dots,\bar{X}} co/r \} - Q_{i0} \} \qquad (5.30)$$

In summary, the *algorithm* to solve the reformulated planning modeL(R2) is as follows:

Step 1: Solve (NP)(NT) processing network problems of the form (5.29).

Step 2: Calculate capacity expansion upper bounds through (5.30).

Step 3: Solve the reformulated MILP model (R2).

The following theorem can be established for the tightness of the LP relaxation in Step 3:

Theorem 6. The optimal NPV of the linear programming relaxation of model (R2) is not greater than the optimal NPV of the linear programming relaxation of model (P2), and it may be strictly less.

The proof of the theorem, although in the same spirit, is slightly more complicated than that for Theorems 3 and 4 and is given in Appendix B. The theorem indicates that the new formulation of model (R2) is at least as accurate as that of model (P2), but nothing is said about the degree of its accuracy. However, if the overestimated capacity expansion upper bounds (the ones from (5.30)) are equal to the optimal values of the capacity expansions, the relaxation will yield an integral solution since the formulation of the lot sizing substructures which has been used satisfies Theorem 2. We can then expect that the closer the overestimated values are to the optimal solutions, the more accurate the relaxation will be. Moreover, we anticipate that, for those processes which are profitable, the optimum will be to run them at the highest possible operating level, and therefore the upper bounds from (5.30) will be equal to the optimal values for the capacity expansions in which case the relaxation of model (R2) will be close to an integer solution.

As in the case of model (Rl), due to the reformulation, the relaxation becomes more accurate but the number of continuous variables and constraints of the model is at the same time increased. This increase is polynomial in the number of time periods (NT) and the number of processes (NP) since we have added $(NP)(NT)^2(NT+l)/2$ new variables and $(NP)(NT)^2(NT\text{-}i\text{-}l)\text{-}(NP)(NT)$ new constraints to the original model (P2).

6. Computational Results

Eight scheduling example problems will be considered as shown in Table 2. Examples BATCH5, 6, 7 and 8 were derived from examples BATCH1, 2, 3 and 4, respectively, by increasing the demands by 50%. The state task networks for examples BATCH1 to 4 are shown in Figures 4, 8, 9 and 10 and they are taken from Kondili (1987). The data used are given in Tables 1, and 3 to 5. The mass balance coefficients which are different than 1 are shown on the state task networks. Also, ten planning examples will be considered as shown in Table 6. These examples are taken from Sahinidis and Grossmann (1989). All the 18 test problems were solved through the modelling system GAMS (Brooke *et al.,* 1988). The procedure was executed on an IBM-3090 and MPSX-MIP/370 (IBM, 1988) was used to solve the MILPs.

Computational results using branch and bound to solve the MILP models (PI), (P2), (Rl) and (R2) for our 18 test problems are shown in Tables 7 through 11. The effect of the reformulation on the problem size is shown in Table 7. The number of continuous variables and constraints is increased, but as pointed out in previous sections this increase is polynomial in size.

Table 8 shows the effect of the reformulation on the linear programming relaxation of the problems. Total profit (sales revenue minus total cost) is shown for the scheduling problems while the net present value is shown for the planning problems. After the reformulation, the relaxation becomes tighter in the sense that the gap between the integer solution and the relaxation is considerably reduced (between 3% and 100%).

Table 9 shows the effect of the reformulation on the computational requirements of the solution. For all the test problems, branch and bound has now to examine a much smaller number of nodes.

For the scheduling problem, the CPU times are reduced in all cases (the reductions are between 10% and 80%). By comparing examples BATCH1, 2, 3, and 4 to examples BATCH5, 6, 7 and 8, respectively, we observe that as the demand increases the effect of the reformulation becomes less important since the LP relaxation gap of the standard formulation becomes fairly small (the problem becomes easier). Actually, computational results have indicated that for sufficient large demands the standard formulation can be solved so effectively (almost as an LP) that the reformulation requires larger CPU times since it involves more variables and constraints.

Although there is no effect on the CPU requirements for the small planning problems, note that the CPU times for the larger examples are up to one order of magnitude lower than those with the conventional model. For instance, in problem PLAN9 the reduction is from 35 minutes to only 3 minutes. Similarly, problems PLAN6, 7, and 10 exhibit significant reductions. Moreover, the reformulation makes possible the solution to optimality of problem PLAN8 in less than 4 minutes; this is an example which could not be solved after 92 minutes with the original formulation. The CPU times in Table 9 include the time needed to solve the linear programs to evaluate the upper bounds for the reformulation variables of the planning model (R2). However, this time is small when compared to the total. For example, for the largest problem this is less than 10 seconds for all the 156 LPs (using MINOS and *not* any specialized algorithm). For the rest of the problems, this time is almost zero. Some statistics for these LPs are shown in Table 10.

By comparing the CPU time reductions in Table 9 we observe that for the more computationally intensive examples (BATCH2, 4, 6, 8 and PLAN8, 9, 10) the effect of the reformulation on the planning problems is more substantial. This is due to the fact that the reduction of the LP relaxation gap is larger for the planning problems (Table 8).

Finally, the entries of Table 11 have been calculated from the number of iterations and the number of nodes presented in Table 9. As seen in Table 11, the average number of iterations per node of the branch and bound tree is larger in the case of the reformulated models. This apparently happens because the reformulation introduces extra variables and constraints. This observation, coupled with the fact that the increase in the number of variables and constraints is, respectively, $O(NT^2)$ and $O(NT^3)$, points out that the reformulation is expected to be more effective in problems which involve a small rather than a large number of time periods.

7. Conclusions

The results of this paper have been based on the observation that multiperiod planning and scheduling problems reduce to lot sizing problems when a subset of the variables are fixed (eg. production, purchases, sales). To take advantage of this property, a variable disaggregation technique has been proposed for reformulating conventional MILP models for these problems. The reformulation strategy was applied to a batch scheduling problem and to the problem of long range planning for capacity expansion of chemical complexes. In both cases, the reformulation led to MILPs with tighter linear programming relaxation which for large problems gave solution time reductions of up to one order of magnitude, when compared to the solution requirements of the conventional formulations of these problems. We anticipate that the reformulation strategy proposed in this paper can be applied to a large class of multiperiod and multistage production planning and scheduling problems in the chemical industries.

Acknowledgments

The authors would like to acknowledge partial financial support from the National Science Foundation under Grant CBT-8908735. Access to the IBM-3090 computer at the Cornell Theory Center is also gratefully acknowledged.

REFERENCES

Balas, E., "Disjunctive Programming and a Hierarchy of Relaxations for Discrete Optimization Problems", *SIAM J. Alg. Disc. Math.*, 6(3), 466-486 (1985).

Barany, I., T.J. Van Roy, and L.A. Wolsey, "Strong Formulations for Multi-item Capacitated Lot Sizing", *Management Science*, 30(10), 1255-1261 (1984).

Beale, E.M.L., "Branch and Bound Methods for Mathematical Programming Systems", *Annals of Discrete Mathematics*, 5, 201-210 (1979).

Brooke, A., D. Kendrick and A. Meeraus, *GAMS - A User's Guide*, The Scientific Press, Redwood City (1988).

Chen, C.J., and M. Enguist, "A Primal Simplex Approach to Pure Processing Networks", *Management Science*, 32(12), 1582-1598 (1988).

IBM, *Mathematical Programming System Extended 1370 (MPSX 1370) Version 2*, Program Reference Manual, Rome (1988).

Jeroslow, R.G., and J.K. Lowe, "Modelling with Integer Variables", *Mathematical Programming Study*, 22, 167-184 (1984).

Jeroslow, R.G., and J.K. Lowe, "Experimental Results on the New Techniques for Integer Programming Formulations", */. Opl. Res. Soc*, 36(5), 393-403 (1985).

Koene, J., *Minimal cost flow in processing networks: a primal approach*, Centrum voor Wiskunde en Informatica, Amsterdam (1983).

Kondili, E., *Optimal Scheduling of Batch Processes*, Ph.D. Thesis, Imperial College of Science and Technology, London (1987).

Kondili, E., C. C. Pantelides, and R. W. H. Sargent, "A General Algorithm for Scheduling Batch Operations", to appear in *Computers and Chemical Engineering* (1990).

Krarup, J., and O. Bilde, "Plant Location, Set Covering and Economic Lot Size: An O(mn) Algorithm for Structured Problems", in L. Collatz et al. (eds.), *Optimierung bei graphentheoretischen und ganzzahligen Problemen*, International Series of Numerical Mathematics, 36, 155-180, Birkhäuser Verlag, Basel (1977).

Ku, H.M., D. Rajagopalan, and I. Karimi, "Scheduling in Batch Processes", *Chemical Engineering Science*, 83(8), 35-45 (1987).

Martin, R.K., "Generating Alternative Mixed-Integer Programming Models Using Variable Redefinition", *Operations Research*, 35(6), 820-831 (1987a).

Martin, R.K., "Using Separation Algorithms to Generate Mixed Integer Model Reformulations", Paper presented at the *TIMSIORSA meeting*, (April 1987b).

McBride, R.D., "Solving Embedded Generalized Network Problems", *European Journal of Operational Research,* 21, 82-92 (1985).

Nemhauser, G.L., and L.A. Wolsey, *Integer and Combinatorial Optimization,* Wiley, New York (1988).

Pochet, Y., and L.A. Wolsey, "Lot-Size Models with Backlogging: Strong Reformulations and Cutting Planes", *Mathematical Programming,* 40(3), 317-335 (1988).

Rardin, R.L., and U. Choe, *Tighter Relaxations of Fixed Charge Network Flow Problems,* Georgia Institute of Technology, Industrial and Systems Engineering Report Series #J-79-18, Atlanta (1979).

Reklaitis, G. V., "Progress and Issues in Computer Aided Batch Process Design", paper presented at *Foundations of Computer Aided Process Design 1989,* Snowmass Village, Colorado (1989).

Rippin, D. W. T., "Simulation of Single- and Multiproduct Batch Chemical Plants for Optimal Design and Operation", *Computers and Chemical Engineering,* 7 (3), 137-156 (1983).

Robertson, J.M., "Plan Small for Expansion", *Chemical Engineering Progress,* 63(9), 87-90 (1967).

Robertson, $J.M_M$ "Design for Expansion. Part 1: Economics", *Chemical Engineering,* pp 87-90, April 22 (1968a).

Robertson, J.M., "Design for Expansion. Part 1: Engineering Optimization", *Chemical Engineering,* pp 187-194, May 6 (1968b).

Sahinidis, N.V., and I.E. Grossmann, "Reformulation of the Multiperiod MILP Model for Capacity Expansion of Chemical Processes", *submitted for publication* (1989).

Sahinidis, N.V., I.E. Grossmann, R. Fornari and M. Chathrathi, "Long Range Planning Model for the Chemical Process Industries", *Computers and Chemical Engineering,* 13 (9), 1049-1063 (1989).

Tedder, D.W., "Optimization for Energy Conservation", *Chemical Engineering Progress,* 63-67, (August 1980).

Twaddle, W.W., and J.B. Malloy, "Evaluating and Sizing New Chemical Plants", *Chemical Engineering Progress,* 62(7), 90-96.(1966).

Wagner, H.M., and T.M. Whitin, "Dynamic Version of the Economic Lot Size Model", *Management Science,* 5, 89-96 (1958).

APPENDIX A: Proof of Theorem 5

Theorem 5. Problems (P3-i) and (P4-i) have the same optimal solution.

Proof: We shall show that (P3-i) and (P4-i) have the same set of feasible solutions. Note first of all, that by summing the equality constraints in (5.14), one can solve for Q_{it}. Then the result can be substituted into (5.15) therefore eliminating the variables Q_{it} and the equality constraints (5.14) from model (P3-i). In this case, (5.15) becomes:

$$Q_{i0} + \sum_{T=1}^{t} QE_{iT} \geq W_{m_i t} \qquad t = 1,NT \qquad \text{(A-1)}$$

Similarly, in model (P4-i), one can solve (5.20) for SQ_{it} and substitute the result into the nonnegativity constraint SQ_{it} (5.16). Then (5.20) and SQ_{it} can be eliminated by rewriting the nonnegativity constraint as follows:

$$\sum_{T=1}^{t} QE_{iT} \geq \sum_{T=1}^{t} d_{it} \qquad t = 1,NT \qquad \text{(A-2)}$$

We need to prove that feasibility in (A-1) implies feasibility in (A-2) and *vice versa*. In the following, we drop the indices i and m_i for simplicity; so consider any process i. The case where none of the flows W_t (t=1, NT) exceeds the installed capacity is trivial since no expansions are required for both problems. Consider the case of arbitrary flows where expansions are required and let p_1 be the earliest time period for which $W_{p_1} > Q_0$. Also let $p_2 > p_1$ be the earliest time period for which $W_{p_2} > W_{p_1}$. Continue in this way to define the set of time periods $N_p = \{p_1, p_2, p_3, ..., p_n\}$ for which $p_1 < p_2 < p_3 < ... < p_n$ and

$$Q_0 < W_{p_1} < W_{p_2} < ... < W_{p_{n-1}} < W_{p_n} \qquad \text{(A-3a)}$$

Because of the way N_p is constructed, we also have:

$$W_p \leq W_{p_\tau} \qquad \text{if } p_\tau < p < p_{\tau+1}, \text{ with } p_\tau, p_{\tau+1} \in N_p,\ p \notin N_p \qquad \text{(A-3b)}$$

From the definitions (5.22) to (5.24):

$$d_{p_1} = W_{p_1} - Q_0, \qquad d_{p_2} = W_{p_2} - W_{p_1}, \qquad d_{p_3} = W_{p_3} - W_{p_2},$$

$$\ldots, \quad d_{p_n} = W_{p_n} - W_{p_{n-1}} \qquad \text{while } d_p = 0 \text{ for } p \notin N_p \tag{A-4}$$

For any time period p ($1 \le p \le NT$), we have:

$$\sum_{t=1}^{p} d_t = d_{p_1} + d_{p_2} + \ldots + d_{p_k} \tag{A-5}$$

where k is the largest element of N_p not exceeding p. Substituting (A-4) into (A-5) yields:

$$\sum_{t=1}^{p} d_t = W_{p_k} - Q_0 \tag{A-6}$$

Then for any point feasible in (P3-i) we have

$$\sum_{t=1}^{p} QE_t \ge W_{p_k} - Q_0 = \sum_{t=1}^{p} d_t \tag{A-7}$$

where the inequality follows from (5.25) and the equality from (A-6). Since constraint (A-7) implies (A-2), it follows that for any capacity expansion sequence which is feasible in problem (P3-i), the demand of problem (P4-i) will be satisfied for any period p (p=l, NT).

Inversely, for any capacity expansion sequence satisfying the demand of problem (P4-i) and for any time period p (p=l, NT), we have:

$$\sum_{t=1}^{p} QE_t \ge \sum_{t=1}^{p} d_t = W_{p_k} - Q_0 \ge W_p - Q_0 \tag{A-8}$$

where the first inequality follows from the feasibility of problem (P4-i) (constraint (3-15)), the equality from (A-6), and the second inequality from (A-3) and the definition of k in (A-5). Since (A-8) implies (5.25), it follows that any feasible point in (P4-i) corresponds to a feasible point in (P3-i).

Since the problems (P3-i) and (P4-i) have the same set of feasible solutions and they have the same objective function, they also have the same optimal solution. •

APPENDIX B: Proof of Theorem 6

Theorem 6. The optimal cost of the linear programming relaxation of model (R2) is not greater than the optimal cost of the linear programming relaxation of model (P2), and it may be strictly less.

Proof: First we observe that constraints (53) can be used to solve for the variables Q_{tr} of model (P2) and then both these variables and constraints can be eliminated with the provision that (5.4) is changed to:

$$Q/0+ \sum_{x=1}^{t} QE/t \;*\; W_{m_t}, \qquad (5.40$$

Now with the exception of (5.4*) the rest of the constraints of model (P2) also appear in model (R2). But from (5.25):

$$Q/o + \sum_{x=1}^{t} OP* \;*\; Q;o + \sum_{x=1}^{t} 9_{to}$$

This means that (5.40 is implied by (5.27). It follows that every solution to the linear programming relaxation of model (R2) gives rise to a feasible solution of the linear programming relaxation of model (P2). This shows that the optimal net present value of the linear programming relaxation of (R2) cannot be greater than that of the linear programming relaxation of (P2). The examples of Section 6 show that the linear programming relaxation of (R2) can yield a strictly smaller upper bound, thus completing the proof. •

LIST OF TABLES

Table 1: Data used for example BATCH1.

Units - Tasks

Units	Size	Units Suitability	Processing times
Unit 1	1500	Task 1	1
Unit 2	1000	Task 2	1
Unit 3	1000	Task 3	1

States

States	Capacity Limits	Prices
State 1 (Feed)	unlimited	5
State 2 (Intermediate)	5000	
State 3 (Product 1)	unlimited	10
State 4 (Product 2)	unlimited	8

Demands (S_{st})

s \ t	4	6	7	10	11	12
Product 1	200		300	400	100	
Product 2	50	150		200		100

Cost Data

$\alpha_{ijt} = 200$; $\beta_{ijt} = 0.6$; $\gamma_{st} = 0.18$

Table 2: **The scheduling example problems.**

Example	States	Tasks	Units	Time Periods
BATCH1, 5	4	3	3	12
BATCH2, 6	9	5	4	10
BATCH3,7	9	6	4	8
BATCH4, 8	13	8	6	8

Table 3: Data used for example BATCH2.

Units - Tasks

Units	Size	Units Suitability	Processing times
Heater	100	Heating	1
Reactor 1	50	Reactions 1,2,3	2,2,1
Reactor 2	80	Reactions 1,2, 3	2,2,1
Still	200	Separation	1 for Product 2, 2 for Intermediate AB

States

States	Capacity Limits	Prices
Feeds A, B, C	unlimited	0
Hot A	100	
Intermediate AB	200	
Intermediate BC	150	
Intermediate E	200	
Product 1	unlimited	60
Product 2	unlimited	45

Demands ($S_s j$)

t	5	**6**	**8**	9	10
Product 1	20	10	20	12	
Product 2				32.5	32.5

Cost Data

a.. = 20 ;

$\beta_{11t} = 0.1$; $\beta_{23t} = 0.25$; $\beta_{33t} = 0.25$; $p_{43t} = 0.15$;

$_{22t} = 0.16$; $h2t = 0.35$; $*42i = 0.1$; $l_{st} = 2.1$;

Table 4: Data used for example BATCH3.

Units - Tasks

Units	Size	Units Suitability	Processing times
Unitl	2029	Task A1	2
Unit 2	1690	Tasks A2.C1	1,1
Unit 3	720	Tasks B1,C2	1,2
Unit 4	929	Task B2	1

States

States	Capacity Limits	**Prices**
Feeds A, B, C	unlimited	**0**
Intermediate A	2500	
Intermediate B	unlimited	
Intermediate C	unlimited	
Product A	unlimited	80
Product B	unlimited	90
Product C	unlimited	100

Demands ($S_s j$)

s \ t	3	4	**5**	6	7	8
Product A		169	33.9		169	**169**
ProductB	92.9	92.9	92.9	92.9	92.9	92.9
ProductC						144

Cost Data

α_{ijt} = **200** ; |V = 0-25 ; γ_{st} =0-1

Table 5: Data used for example BATCH4.

Units - Tasks

Units	Size	Units Suitability	Processing times
Unit 1	1000	Task 1	1
Unit 2	2500	Tasks 3, 7	1
Unit 3	3500	Task 4	1
Unit 4	1500	Task 2	1
Unit 5	1000	Task 6	1
Unit 6	4000	Tasks 5, 8	1

States

States	Capacity Limits	Prices
Feeds 1, 2, 3	unlimited	0
Intermediate 4	1000	
Intermediate 5	1000	
Intermediate 6	1500	
Intermediate 7	2000	
Intermediate 8	0	
Intermediate 9	3000	
Products 1, 2, 3, 4	unlimited	18, 19, 20, 21

Demands (S_{st})

s \ t	3	4	5	6	7	8
Product 1	110	110	133.3	100	33.3	33.3
Product 2		233.1	260		360	360
Product 3			116.6	56.6		116.6
Product 4				333.3	333.3	685.8

Cost Data

$\alpha_{ijt} = 20$; $\beta_{ijt} = 0.55$; $\gamma_{st} = 0.1$

Table 6: The planning example problems.

Example	Processes	Time Periods	Chemicals
PLAN1, 2, 3, 4	3	3	3
PLAN5, 6, 7	10	4	6
PLAN8, 9, 10	38	4	25

Table 7: Effect of the reformulation on the problem size.

Example	Initial Model: Constraints	Initial Model: Variables Total	Initial Model: Variables Integer	Initial Model: Nonzeroes	Reformulation: Constraints	Reformulation: Variables Total	Reformulation: Variables Integer	Reformulation: Nonzeroes
BATCH1, 5	90	133	36	421	154	211	36	721
BATCH2, 6	298	281	80	1113	366	326	80	1518
BATCH3,7	152	193	48	601	211	282	48	895
BATCH4, 8	183	257	64	809	271	322	64	1337
PLAN1, 2, 3	49	55	9	160	76	64	9	217
PLAN4	46	55	9	142	73	64	9	199
PLAN5	195	225	40	639	355	285	40	989
PLAN6	185	225	40	599	345	285	40	949
PLAN7	199	225	40	719	359	285	40	1,069
PLAN8	785	961	152	2,551	1,431	1,189	152	4,033
PLAN9	823	961	152	2,703	1,469	1,189	152	4,185
PLAN10	827	961	152	3,007	1,473	1,189	152	4,489

Table 8: Effect of the reformulation on the linear programming relaxation.

Example	Integer optimum	Initial Model		Reformulation		
		Relaxation optimum	Gap	Relaxation optimum	Gap	Gap reduction
	Z_{IP}	Z_R	$\frac{Z_R}{Z_{IP}} \times 100$	Z_R^R	$\frac{Z_R^R}{Z_{IP}} \times 100$	$\frac{Z_R - Z_R^R}{Z_R - Z_{IP}} \times 100$
BATCH1	3,230	4,200	130.0	3,880	120.1	33
BATCH2	5,593	5,976	106.9	5,943	106.3	9
BATCH3	105,756	106,768	101.0	106,238	100.5	52
BATCH4	60,533	60,933	100.7	60,859	100.5	19
BATCH5	5,445	6,300	115.7	6,086	111.8	25
BATCH6	8,239	8,921	108.3	8,898	108.0	3
BATCH7	158,971	160,163	100.8	159,725	100.5	37
BATCH8	91,041	91,441	100.5	91,382	100.4	15
PLAN1	1,697	1,898	111.8	1,744	102.8	77
PLAN2	1,775	1,932	108.8	1,775	100.0	100
PLAN3	1,063	1,246	117.3	1,099	103.4	80
PLAN4	2,235	2,540	113.7	2,305	103.1	77
PLAN5	51,031	51,207	100.3	51,117	100.2	51
PLAN6	51,450	51,837	100.8	51,481	100.1	92
PLAN7	45,248	46,540	102.9	46,370	102.5	13
PLAN8	529.8	648.6	122.5	621	117.2	23
PLAN9	529.8	648.6	122.5	621	117.2	23
PLAN10	529.8	631	119.1	598	112.9	33

Table 9: Effect of the reformulation on the solution of the MILP[a].

Example	Initial Model # nodes	Initial Model #iterations	Initial Model time (sec)	Reformulation # nodes	Reformulation #iterations	Reformulation time[b] (sec)	CPU time reduction (%)
BATCH1	1,140	2,848	8.6	180	1,008	3.5	59
BATCH2	3,899	22,352	112	2635	16,276	97	13
BATCH3	287	1,289	4.7	20	370	0.9	81
BATCH4	10,098	35,467	175	4004	15,096	107	39
BATCH5	321	1,126	3.1	125	828	2.6	16
BATCH6	6,405	35,112	197.4	4,528	25,913	177.7	10
BATCH7	1,268	5,825	20.3	20	1,781	7.7	62
BATCH8	3,382	13,354	57.4	1,750	7,808	49.4	14
PLAN1	10	93	0.6	3	113	0.6	0
PLAN2	14	93	0.6	1	96	0.6	0
PLAN3	11	85	0.6	3	104	0.6	0
PLAN4	11	86	0.6	5	120	0.6	0
PLAN5	37	439	1.7	14	590	1.9	-12
PLAN6	1,064	2,862	10.7	17	544	2	81
PLAN7	1,272	6,305	21.8	23	916	2.7	88
PLAN8	NA[c]	>356,609[c]	>5,520[c]	1,516	14,323	222	>96
PLAN9	28,696	134,440	2,100	1,037	12,329	192	91
PLAN10	4,530	32,713	540	1,164	20,503	324	40

(a) MPSX-MIP/370 computer code used on IBM-3090.

(b) Includes LP computations for upper bounds for the planning problems using MINOS 5.1.

(c) Procedure terminated with a lower bound of 529.8 and an upper bound of 561.

Table 10: Size and number of linear programs solved to obtain upper bounds for model (R2).

Example	Rows	Variables	Nonzeroes	# problems solved
PLAN1, 2, 3, 4	7	9	17	9
PLAN5, 6, 7	17	40	48	40
PLAN8, 9, 10	83	127	236	152

Table 11: Average number of linear programming iterations per node.

Example	Initial Model	Reformulation
BATCH1	2.5	5.6
BATCH2	5.7	6.2
BATCH3	4.5	18.5
BATCH4	3.5	3.8
BATCH5	3.5	6.6
BATCH6	5.5	89
BATCH7	4.6	8.9
BATCH8	3.9	4.5
PLAN1	9.3	37.7
PLAN2	6.6	96
PLAN3	7.7	34.7
PLAN4	7.8	24
PLAN5	11.9	42
PLAN6	2.7	32
PLAN7	5 •	39.8
PLAN8	NA	9.5
PLAN9	4.7	11.9
PLAN10	7.2	17.6

LIST OF FIGURES

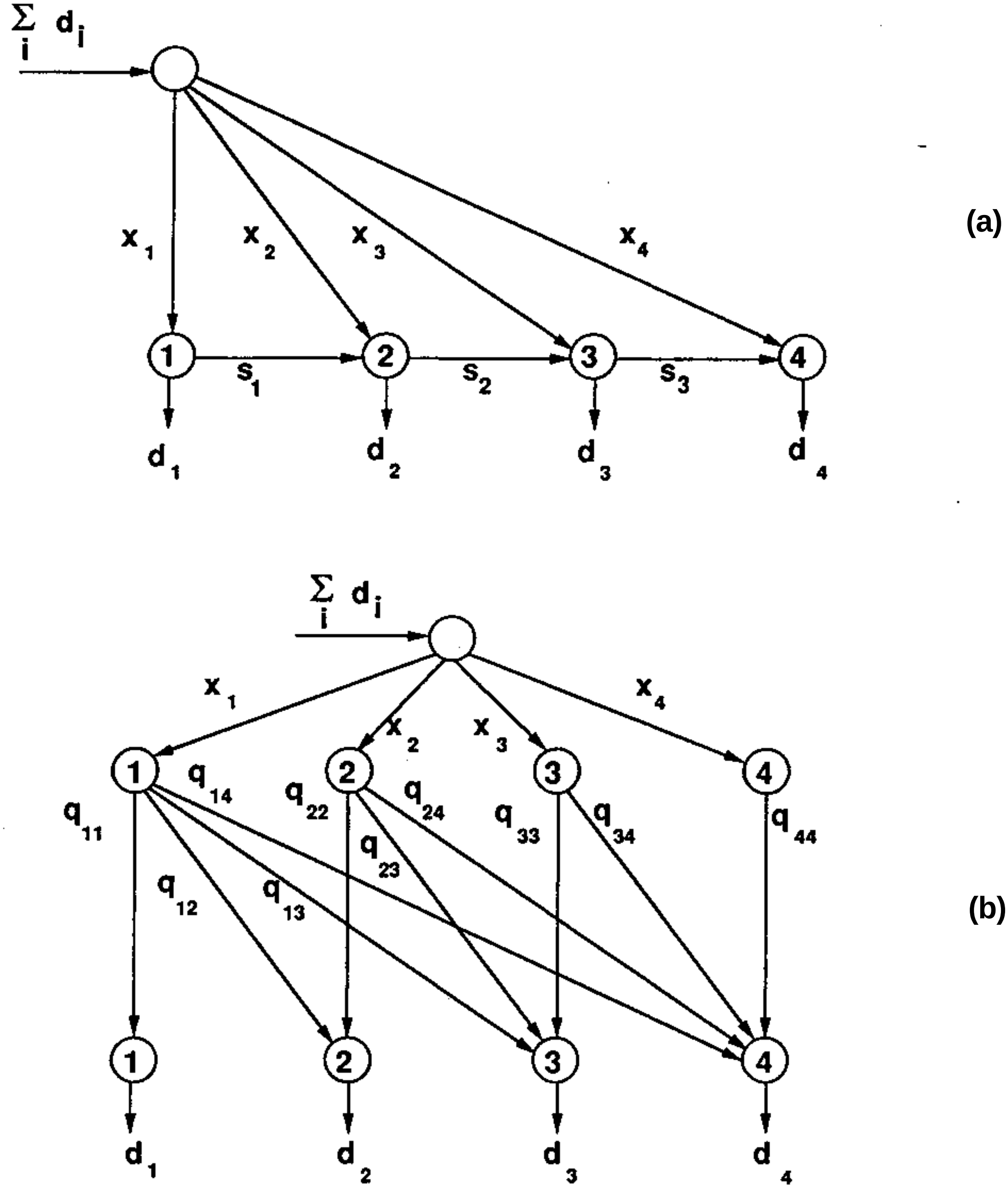


Fig-1: **Lot Sizing Model Representation:**
(a) prior to reformulation
(b) after reformulation

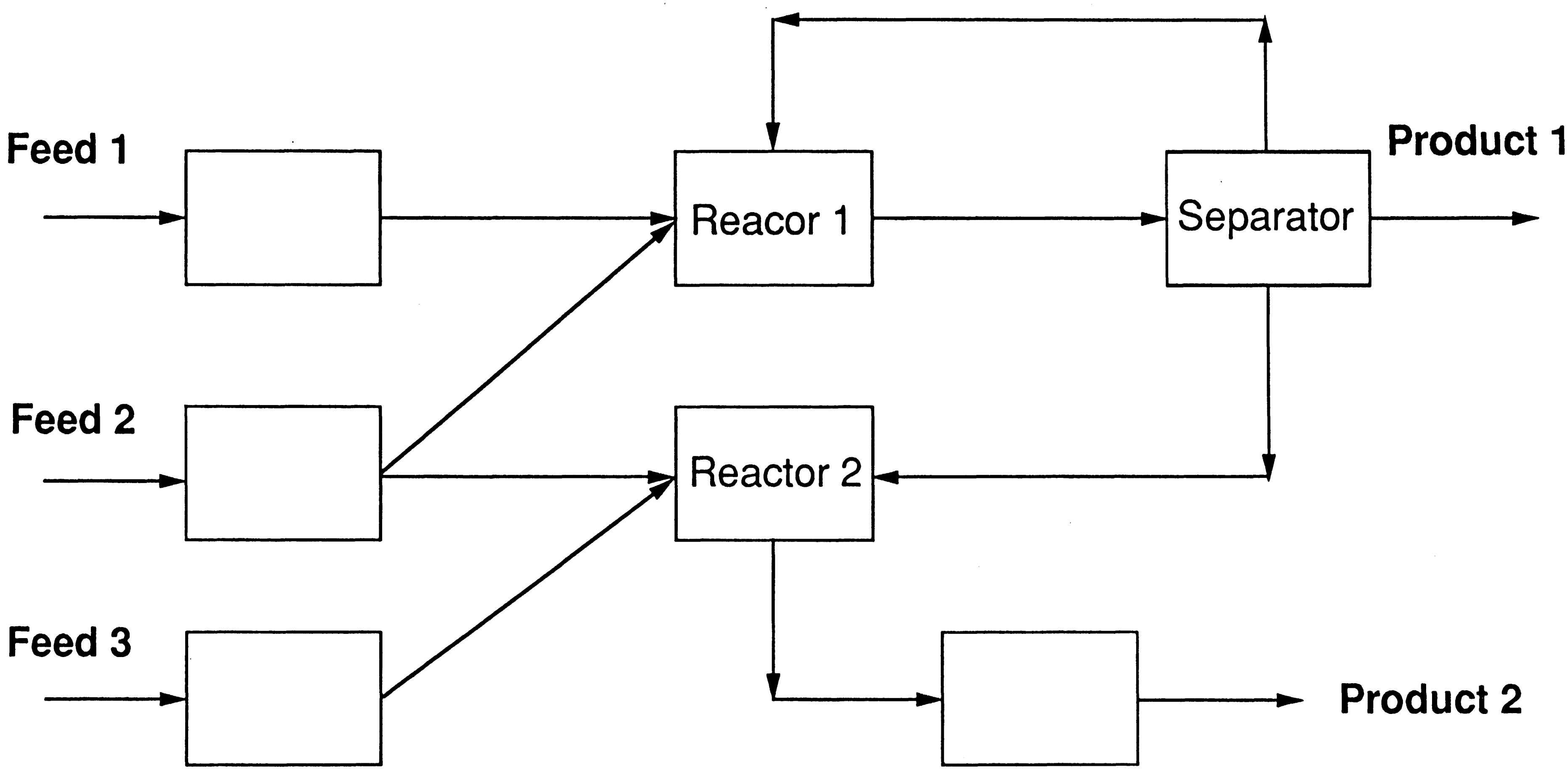


Fig. 2: Conventional Flowsheet Example

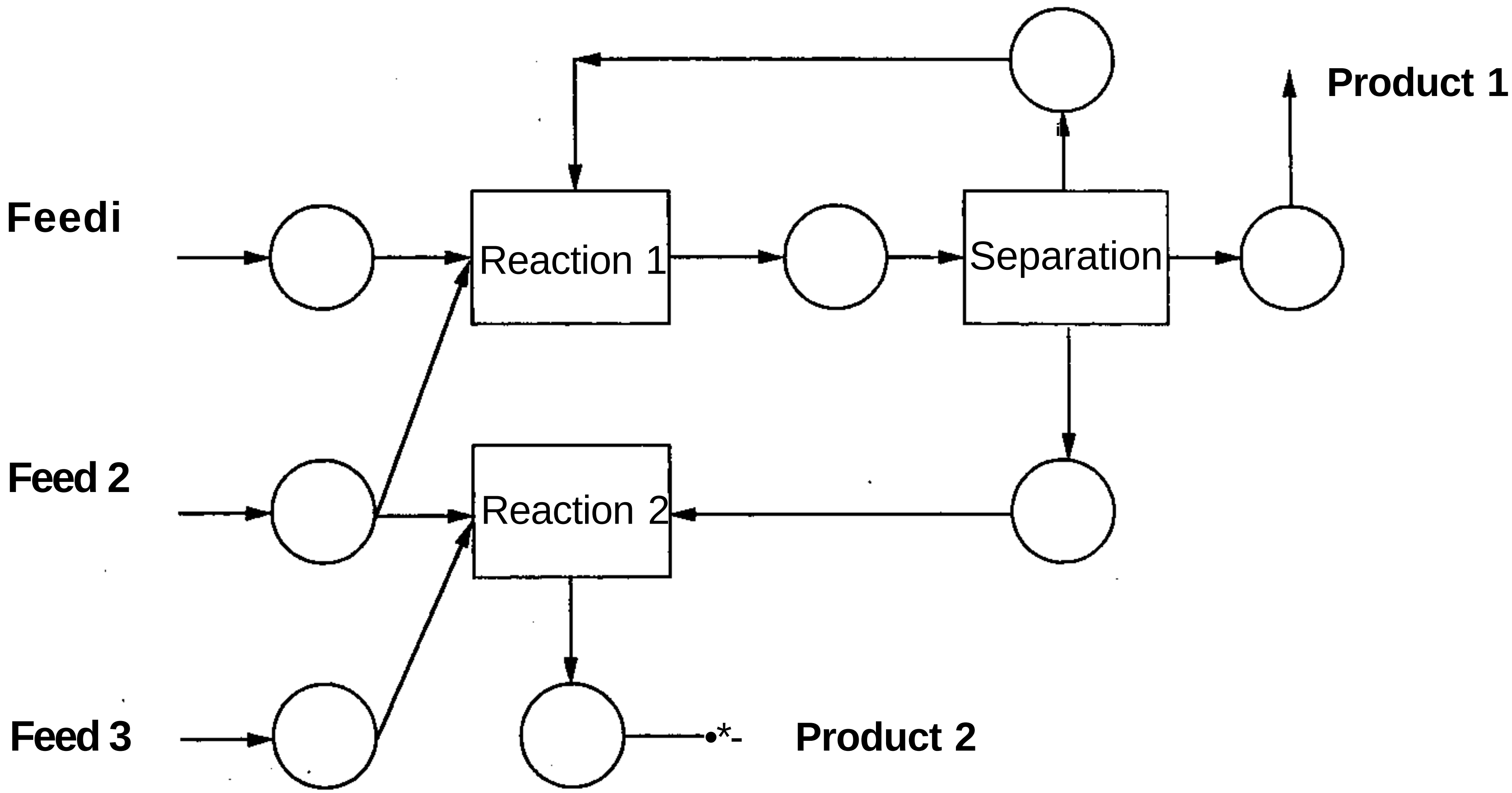


Fig. 3: **State Task Network Example**

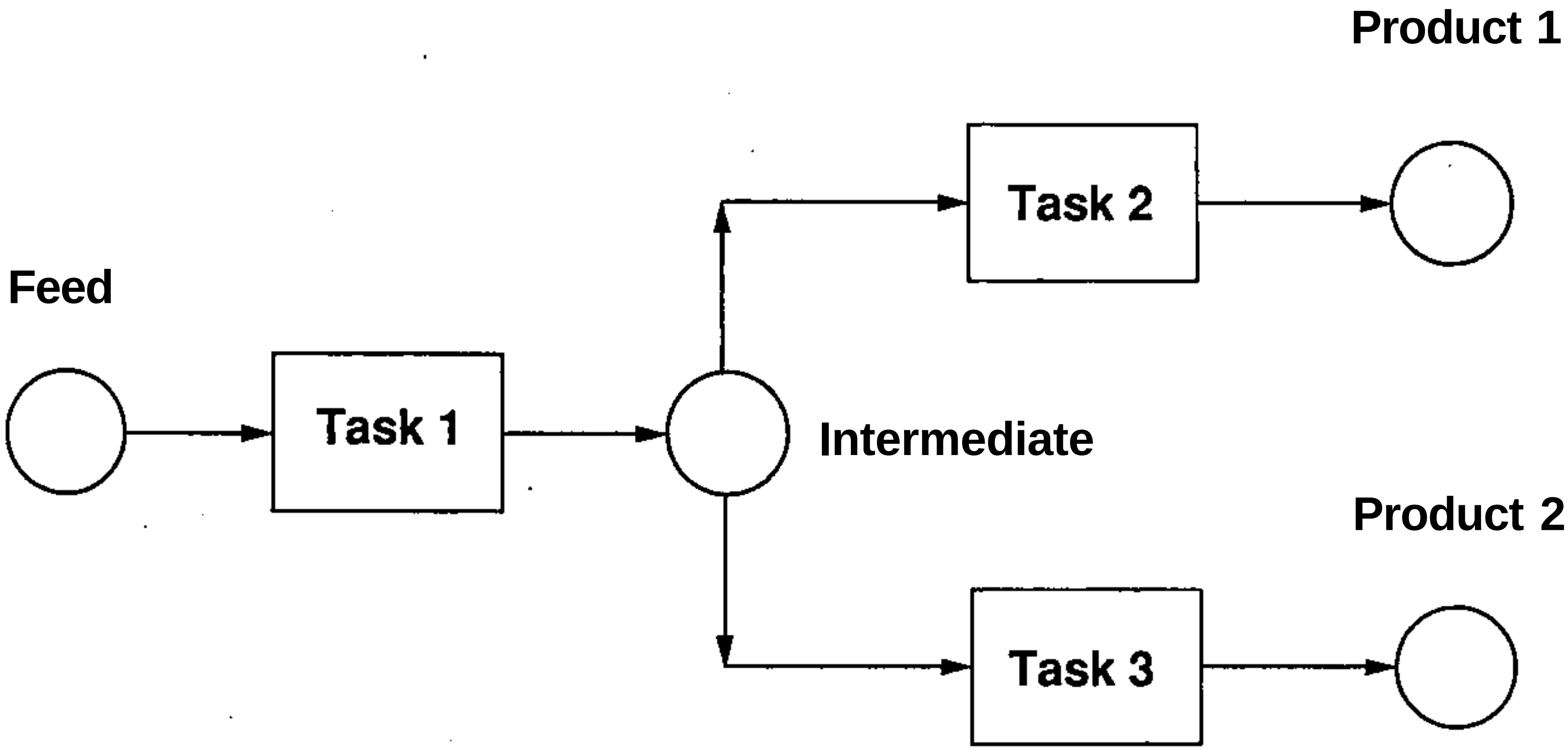


Fig. 4: State Task Network for Problem BATCH1

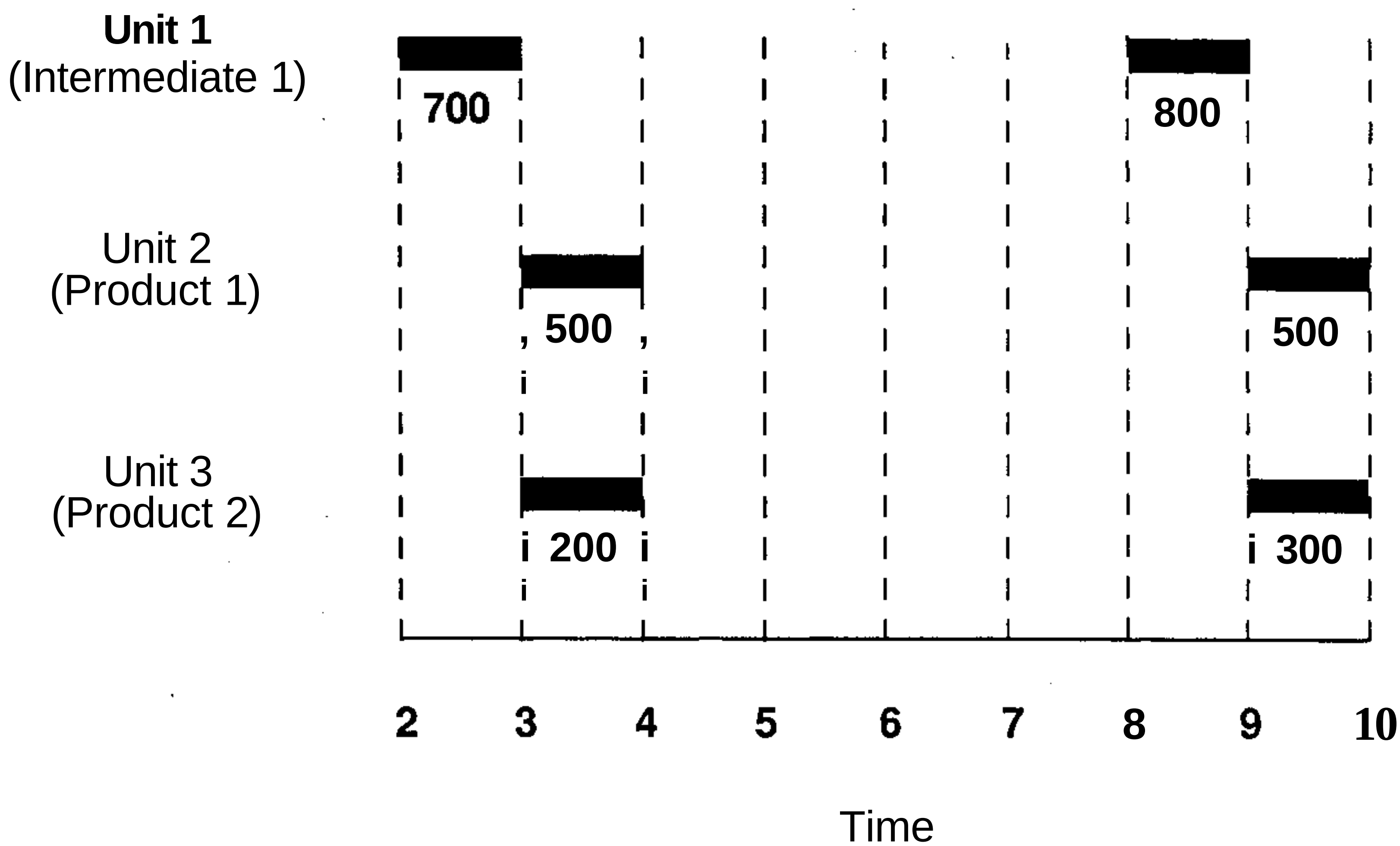


Fig. 5: **Equipment (Product) Schedule for example BATCH1.**

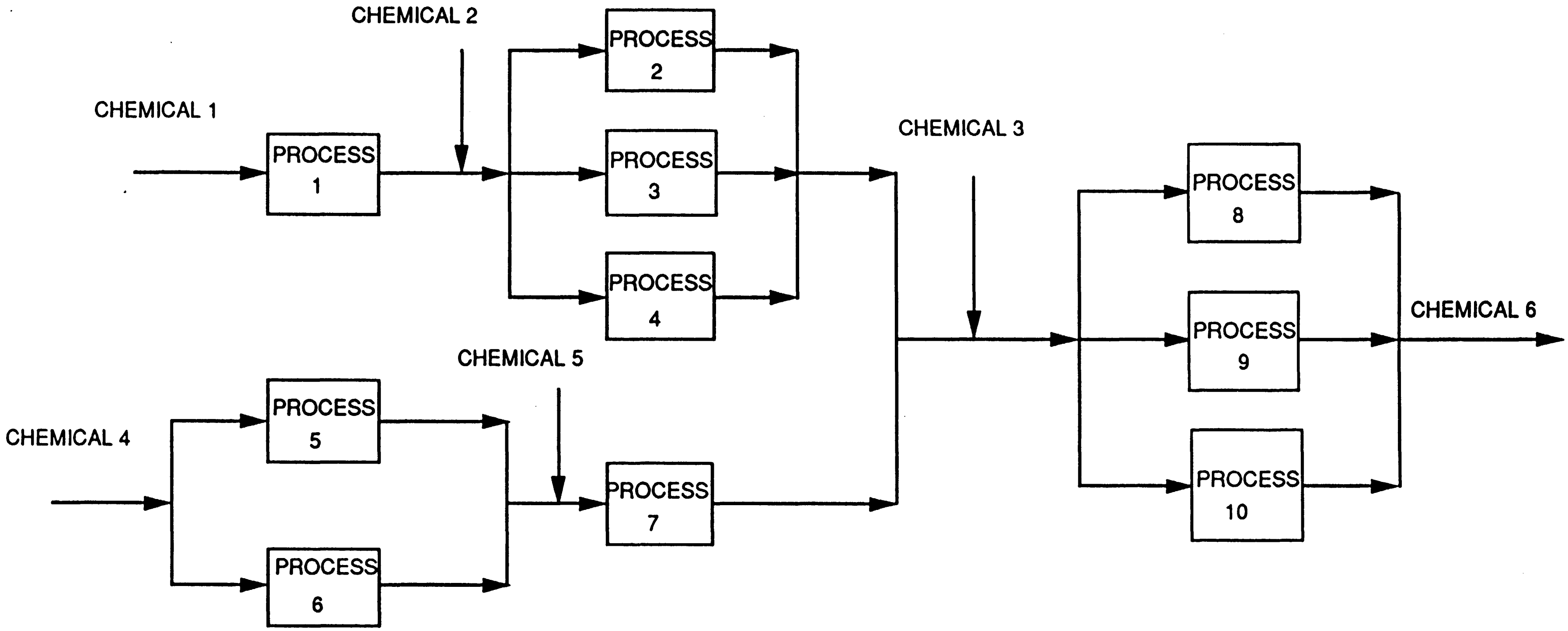
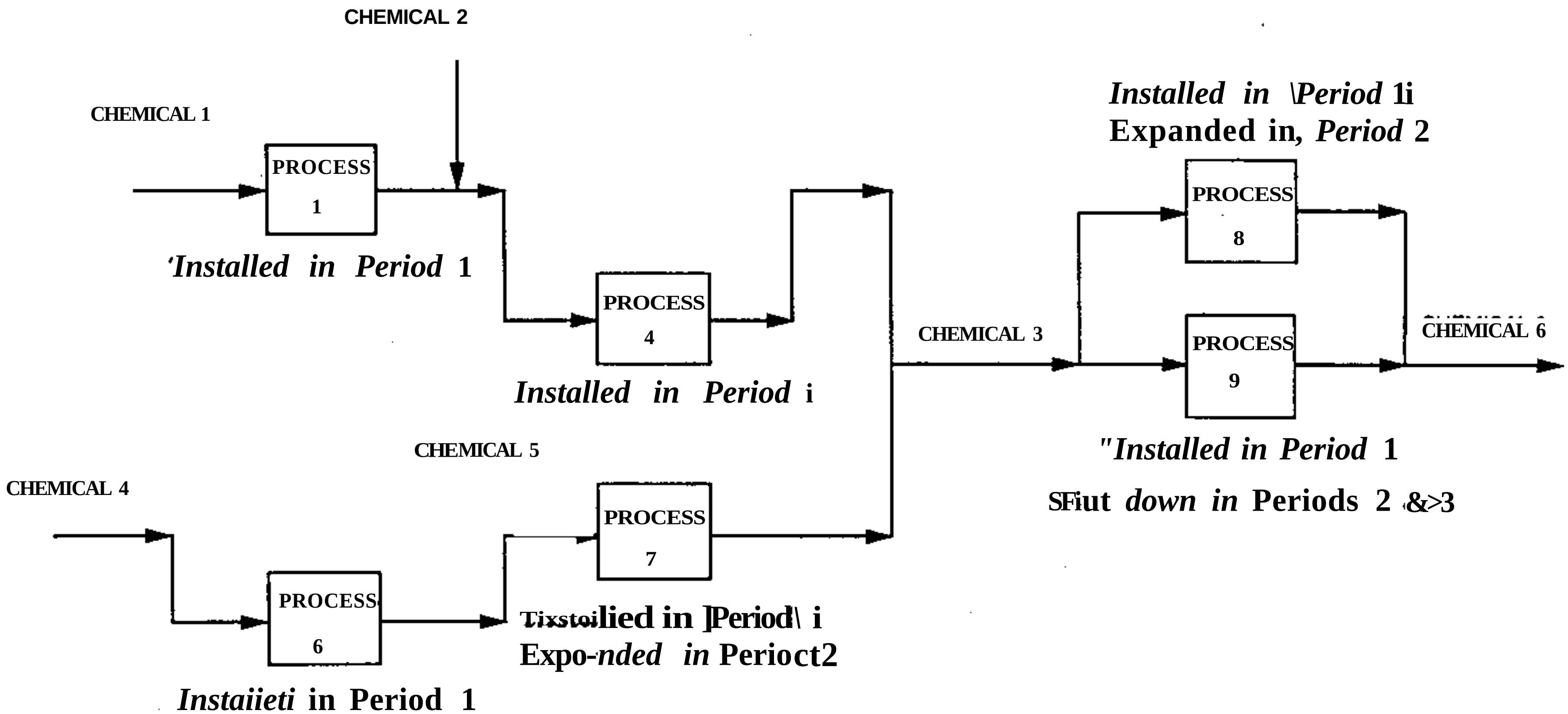


Fig. 6: **Flow Diagram of a Chemical Complex**



<u>Fig. 7:</u> **Optimum Network Configuration**

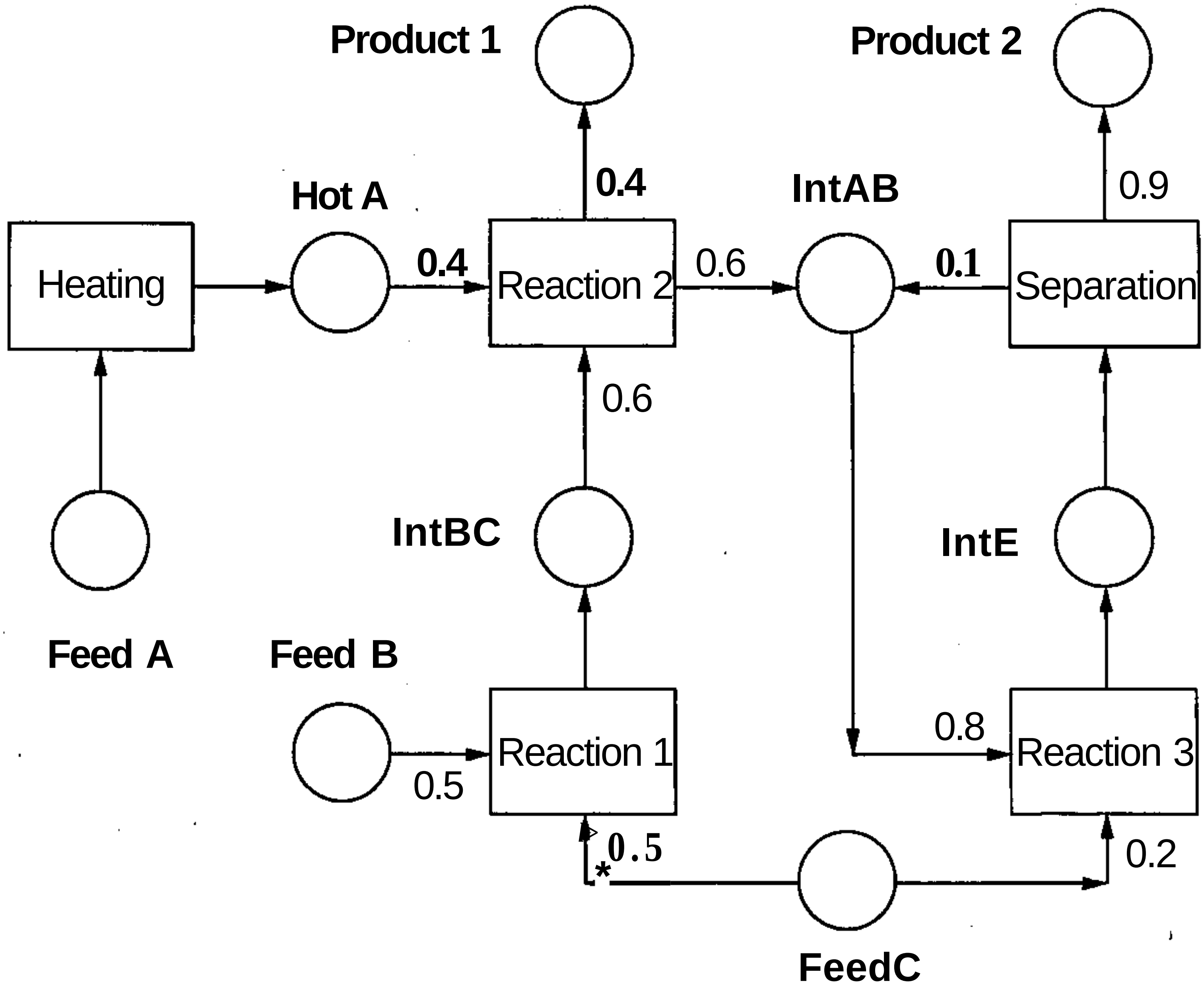
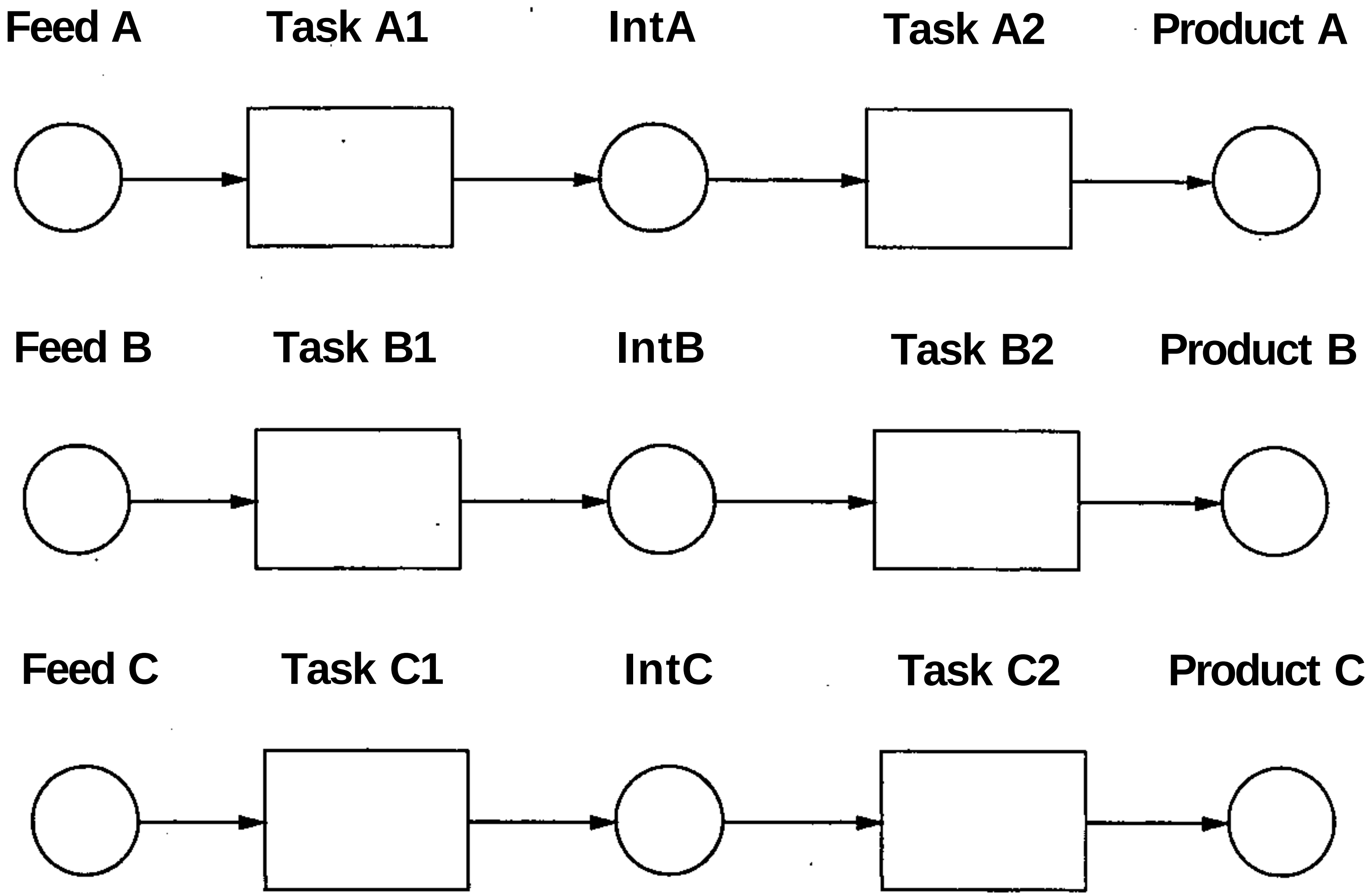
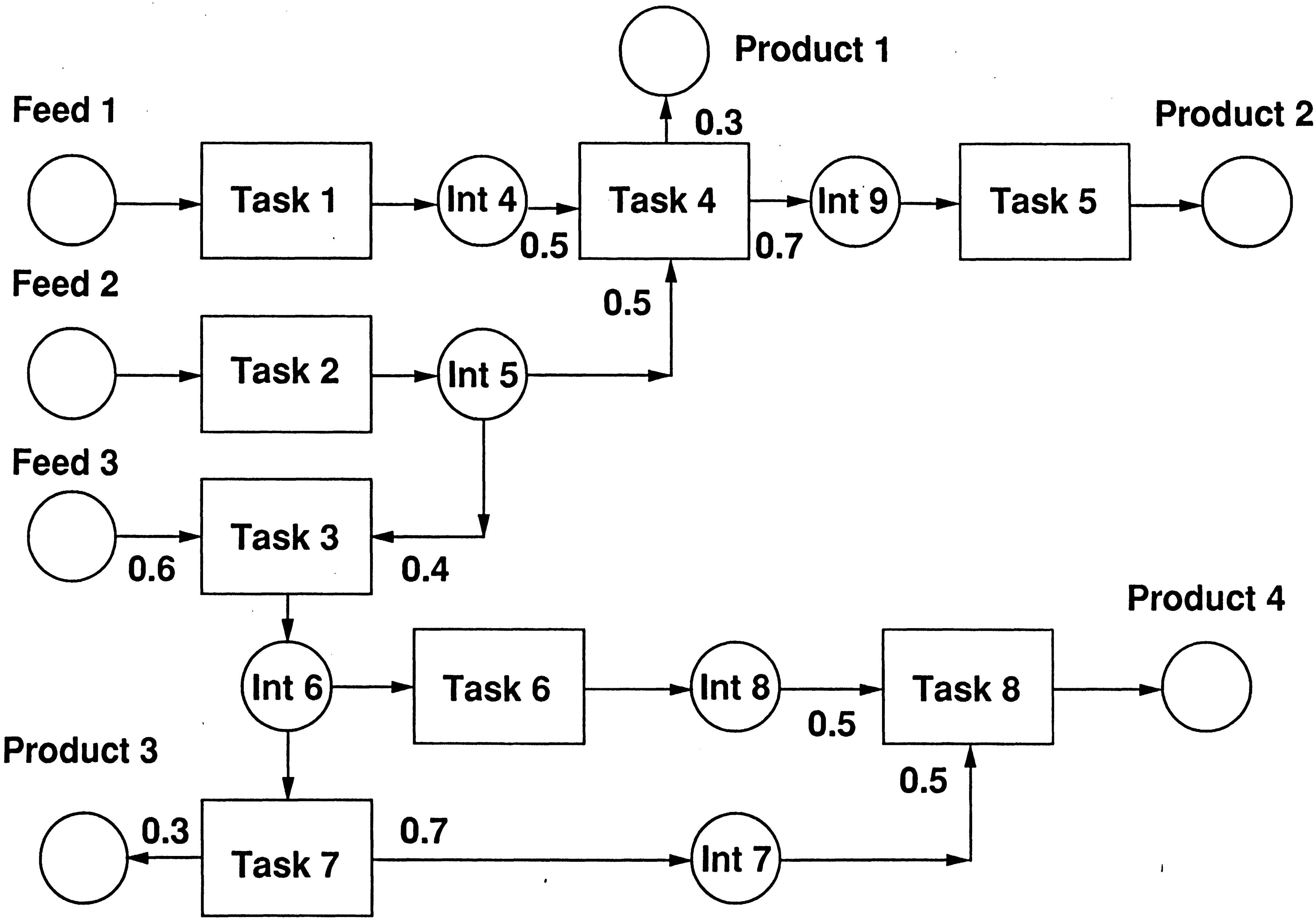


Fig. 8: State Task Network for Problem BATCH2



<u>Fig. 9:</u> **State Task Network for Problem BATCH3**



<u>Fig. 10:</u> State Task Network for Problem BATCH4