Design, Synthesis and Scheduling of Multipurpose Batch Plants via an Effective Continuous-Time Formulation

X. Lin and C. A. Floudas [1]

Department of Chemical Engineering, Princeton University, Princeton, N.J. 08544-5263,USA

Abstract – Design, synthesis and scheduling issues are considered simultaneously for multipurpose batch plants. A previously proposed continuous-time formulation for scheduling is extended to incorporate design and synthesis. Processing recipes are represented by the State-Task Network. The superstructure of all possible plant designs is constructed according to the potential availability of all processing/storage units. The proposed model takes into account the trade-offs between capital costs, revenues and operational flexibility. Computational studies are presented to illustrate the effectiveness of the proposed formulation. Both linear and nonlinear models are included, resulting in MILP and MINLP problems, respectively. The MILP problems are solved using a branch and bound method. Globally optimal solutions are obtained for the nonconvex MINLP problems based on a key property that arises due to the special structure of the resulting models. Comparisons with previous approaches are also presented.

Keywords – Design and Scheduling; Multipurpose Batch Process; Continuous-Time Formulation; MILP; MINLP

1 Introduction

Multipurpose batch plants have been employed extensively for the manufacture of many types of chemicals, particularly those which are produced in small quantities and for which the production processes or the demand pattern are likely to change. In these plants, a wide variety of products

[1]To whom all correspondence should be addressed. Tel: (609) 258-4595; Fax:(609) 258-0211; E-mail: floudas@titan.princeton.edu

can be produced via different processing recipes by sharing available pieces of equipment, raw materials and intermediates, utilities and production time resources. The special inherent operational flexibility represents considerable complexity in the design and synthesis of such plants. There have been several publications in the area of design and operation of multipurpose batch plants (e.g., Grossmann and Sargent (1979); Suhami and Mah (1982); Birewar and Grossmann (1989); Papageorgaki and Reklaitis (1990a,b, 1993); Barbosa-Póvoa and Macchietto (1994); Xia and Macchietto (1997)). In many cases, scheduling strategies are not incorporated or integrated very well, which may lead to over-design or under-design. In order to ensure that any resource incorporated in the design can be used as efficiently as possible, detailed considerations of plant scheduling must be taken into account at the design stage. Therefore, it is important to consider design, synthesis and scheduling simultaneously.

All formulations for design and scheduling of batch processes can be classified into two groups based on the time representations. Examples of discrete time formulations are found in Grossmann and Sargent (1979); Suhami and Mah (1982); Papageorgaki and Reklaitis (1990a,b); Barbosa-Póvoa and Macchietto (1994). Grossmann and Sargent (1979) solved the problem of optimal design of sequential multiproduct batch processes as a mixed-integer nonlinear programming (MINLP) problem. On this basis, Suhami and Mah (1982) studied the optimal design of multipurpose batch plant focusing on a restricted form of the problem as the "unique unit-to-task assignment" case. Papageorgaki and Reklaitis (1990a) criticized many previous formulations in omitting key aspects of the general multipurpose plant, such as alternative assignments of different equipment items to each product task and sharing of the units of the same equipment type among multiple tasks of the same or different products. They proposed a formulation where flexible unit-to-task allocations and non-identical parallel units are considered. A decomposition strategy is also proposed to solve the resulting MINLP problems by Papageorgaki and Reklaitis (1990b). Barbosa-Póvoa and Macchietto (1994) presented a detailed formulation of multipurpose batch plant design and retrofit based on the State-Task Network (STN) description and equally-spaced fixed event time representation proposed by Kondili et al. (1993).

In recent years, attempts have been made to create continuous-time formulations. Xia and Macchietto (1997) presented a formulation based on the variable event time scheduling model of Zhang and Sargent (1996, 1998). A stochastic method is used to solve the resulting nonconvex MINLP problems directly, instead of introducing a large number of auxiliary variables and constraints to reduce the MINLP into a MILP.

Ierapetritou and Floudas (1998a,b); Ierapetritou et al. (1999) proposed a novel continuous-time mathematical model for the general short-term scheduling problem of batch, continuous and semicontinuous processes. It features the key concept of event points and sets of special sequence constraints. Market demands can be specified at the end of the time horizon or within the time horizon with intermediate due dates.

In this paper, we extend the formulation to address the problem of integrated design, synthesis and scheduling of multipurpose batch plants. First, a definition of the problem under consideration is stated, and the representation of problem data is discussed. This is followed by a detailed description of the proposed mathematical formulation. Computational results and comparisons with a previously proposed formulation are also presented.

2 Problem Definition

The integrated design, synthesis and scheduling problem for multipurpose batch plants considered in this paper is stated as follows:

Given

- Production recipes (i.e., the processing times for each task at the suitable units, and the amount of the materials required for the production of each product);
- Potentially available processing/storage equipment and their ranges of capacities;
- Material storage policy;
- Production requirement;

- The time horizon under consideration;

Determine

- The number, type and size of equipment items;
- A feasible operational schedule;
  - The optimal sequence of tasks taking place in each unit;
  - The amount of material being processed at each time in each unit;
  - The processing time of each task in each unit;

so as to optimize a performance criterion, for example, to minimize the capital cost or to maximize the overall profit.

3 Process Recipe and Plant Superstructure Representations

The first stage of solving the design, synthesis and scheduling problem is to develop a general representation of the process recipe. In this work, we employ the concept of State-Task Network(STN) proposed by Kondili et al. (1993). The STN is a directed graph with two types of distinctive nodes: the *state* nodes denoted by a circle and the *task* nodes denoted by a rectangle box. Figure 1 gives an illustration of the STN description of a batch precess named BM process, in which two final products, S5 and S6, are produced from two raw materials, S1 and S2, through four tasks, T1–T4, involving two intermediate materials, S3 and S4.

In addition to the process recipe, the information of potentially available pieces of equipment and their suitability for different tasks is used to construct a superstructure of the plant under consideration that includes all possible designs. For example, based on the BM process recipe in Figure 1 and equipment data in Table 1, we are able to establish a superstructure of the BM plant, as shown in Figure 2. Decisions on the plant structure involve three processing units and one storage vessel. Full connectivity of the processing units/storage vessel network is assumed.

4 Mathematical Formulation

To formulate the mathematical model for integrated design, synthesis and scheduling of multipurpose batch plants, we require the following indices, sets, parameters, and variables:

Indices:

i tasks; j units; s states;

n event points representing the beginning of a task or utilization of a unit.

Sets:

I tasks;

I_j tasks that can be performed in unit (j);

I_s tasks that either produce or consume state (s);

I_p processing tasks;

I_t storage tasks;

J units;

J_i units that can perform task (i);

J_t storage units;

S all involved states;

S_t states that can only be stored in dedicated storage units;

N event points within the time horizon.

Parameters:

ρ^p_{si}, ρ^c_{si} proportions of state (s) produced, consumed by task (i), respectively;

$\alpha_{ij}, \beta_{ij}, \gamma_{ij}$ constant term, coefficient and exponent of variable term of processing time of task (i) in unit (j), respectively;

$\tilde{\alpha}_j, \tilde{\beta}_j, \tilde{\gamma}_j$ constant term, coefficient and exponent of variable term of capital cost of unit (j), respectively;

H time horizon;

p_s price of state (s);

r_s market requirement for state (s) at the end of time horizon;

V_j^{min}, V_j^{max} minimum and maximum possible sizes of unit (j), respectively .

Variables:

$e(j)$ binary variables to determine if unit (j) exists;

$s(j)$ positive variables that determine the size of unit (j);

$wv(i, n)$ binary variables that assign the beginning of task (i) at event point (n);

$yv(j, n)$ binary variables that assign the utilization of unit (j) at event point (n);

$b(i, j, n)$ positive variables that determine the amount of material undertaking task (i) in unit (j) at event point (n);

$d(s, n)$ positive variables that determine the amount of state (s) being delivered to the market at event point (n);

$st(s, n)$ positive variables that determine the amount of state (s) at event point (n);

$t^s(i, j, n)$ time that task (i) starts in unit (j) at event point (n);

$t^f(i, j, n)$ time that task (i) finishes in unit (j) while it starts at event point (n).

Based on this notation the mathematical model involves the following constraints:

Existence Constraints

$$yv(j, n) \le e(j), \quad \forall j \in J, \quad n \in N \tag{1}$$

These constraints express the requirement that a unit can be utilized only if it exists.

Unit Size Constraints

$$V_j^{min} e(j) \leq s(j) \leq V_j^{max} e(j), \quad \forall j \in J \tag{2}$$

These constraints determine the range of the size of each unit. If e(j) equals one, that is, a unit exists, then Constraint (2) corresponds to the lower and upper bounds on the size of the unit, s(j). If e(j) equals zero, then s(j) becomes zero.

Allocation Constraints

$$\sum_{i \in I_j} wv(i,n) = yv(j,n), \quad \forall j \in J, \quad n \in N \tag{3}$$

These constraints express that in each unit (j) and at any event point (n) at most one of the tasks that can be performed in this unit (i.e., $i \in I_j$) should take place. If unit (j) is utilized at event point (n), that is, yv(j,n) equals 1, then one of the wv(i,n) variables should be activated. If unit (j) is not utilized at event point (n), then all corresponding wv(i,n) variables take zero values, that is, no assignments of tasks are made.

Capacity Constraints

$$b(i,j,n) \leq V_j^{max} wv(i,n), \quad \forall i \in I, \quad j \in J_i, \quad n \in N \tag{4}$$

$$b(i,j,n) \leq s(j), \quad \forall i \in I, \quad j \in J_i, \quad n \in N \tag{5}$$

These constraints express the requirement that the batch-size should be within the maximum capacity of a unit (j). If wv(i,n) equals zero, that is, task (i) does not take place at event point (n), then the first constraint enforces b(i,j,n) to be zero. If wv(i,n) equals one, then the second constraint restricts b(i,j,n) to be within the available capacity of unit (j), s(j).

Material Balances

$$st(s,n) = st(s,n-1) - d(s,n) + \sum_{i\in I_s}\rho^p_{si}\sum_{j\in J_i} b(i,j,n-1) + \sum_{i\in I_s}\rho^c_{si}\sum_{j\in J_i} b(i,j,n) \quad \forall s\in S,\ \ n\in N \tag{6}$$

where $\rho^c_{si} \le 0, \rho^p_{si} \ge 0$ represent the proportion of state (s) consumed or produced by task (i), respectively. According to these constraints the amount of material of state (s) at event point (n) is equal to that at event point (n-1) adjusted by any amounts produced or consumed between the event points (n-1) and (n) and the amount required by the market at event point (n) within the time horizon.

Storage Constraints

$$st(s,n) = 0,\ \ \forall s\in S_t,\ \ n\in N \tag{7}$$

These constraints enforce that those states that can only be stored in dedicated storage units have to be consumed by some processing task or storage task immediately after they are produced.

Demand Constraints

$$\sum_{n\in N} d(s,n) \ge r_s,\ \ \forall s\in S \tag{8}$$

These constraints represent the requirement to produce at least as much as required by the market.

Duration Constraints: Processing task

$$t^f(i,j,n) = t^s(i,j,n) + \alpha_{ij} wv(i,n) + \beta_{ij} b(i,j,n)^{\gamma_{ij}}, \;\; \forall i \in I_p, \;\; j \in J_i, \;\; n \in N \tag{9}$$

In Constraints (9), the processing time takes a generally nonlinear form consisting of a fixed term and a variable term depending on the batch-size. When γ_{ij} equal one, these constraints become linear as a special case.

Duration Constraints: Storage task

$$t^f(i,j,n) \geq t^s(i,j,n), \;\; \forall i \in I_t, \;\; j \in J_i, \;\; n \in N \tag{10}$$

$$t^f(i,j,n_{last}) = H, \;\; \forall i \in I_t, \;\; j \in J_i \tag{11}$$

These constraints express that the duration of storage tasks can take any positive value as long as they end at the end of the time horizon. The concept of storage task is introduced to make it possible to treat dedicated storage constraints and processing constraints in a unified and general way.

Sequence Constraints:

Same task in the same unit

$$t^s(i,j,n+1) \geq t^f(i,j,n), \;\; \forall i \in I, \;\; j \in J_i, \;\; n \in N, n \neq n_{last} \tag{12}$$

These constraints state that task (i) starting at event point (n+1) should start after the end of the same task performed in the same unit (j) which has already started at event point (n).

Different tasks in the same unit

$$t^s(i,j,n+1) \geq t^f(i',j,n) - H(1 - wv(i',n))$$
$$\forall j \in J, \;\; i \in I_j, \;\; i' \in I_j, \;\; i \neq i', \;\; n \in N, n \neq n_{last} \tag{13}$$

These constraints are written for tasks (i, i') that are performed in the same unit (j). If both tasks are performed in the same unit they should be at most consecutive. This is expressed by Constraints (13) because if $wv(i', n) = 1$ which means that task (i') takes place at unit (j) at event point (n), then the second term of the right hand side of (13) becomes zero forcing the starting time of task (i) at event point (n+1) to be greater than the end time of task (i') at event point (n); otherwise, the right hand side of (13) becomes negative and the constraint is trivially satisfied.

Different tasks in different units

$$t^s(i,j,n+1) \geq t^f(i',j',n) - H(1 - wv(i',n))$$
$$\forall i, i' \in I, \;\; j \in J_i, j' \in J_{i'}, \;\; j \neq j', \;\; n \in N, n \neq n_{last} \tag{14}$$

Constraints (14) are written for different tasks (i, i') that are performed in different units (j, j') but take place consecutively according to the production recipe. Note that if task (i') takes place in unit (j') at event point (n) (i.e., $wv(i', n) = 1$), then we have $t^s(i, j, n+1) \geq t^f(i', j', n)$ and hence task (i) in unit (j) has to start after the end of task (i') in unit (j').

"Zero-wait" condition

$$t^s(i,j,n+1) \leq t^f(i',j',n) - H(2 - wv(i,n+1) - wv(i',n))$$
$$\forall i, i' \in I, \;\; j \in J_i, j' \in J_{i'}, \;\; n \in N, n \neq n_{last} \tag{15}$$

Constraints (15) are written for different tasks (i, i') that take place consecutively with "zero-wait" condition due to storage restrictions on the intermediate material. Combined with Constraints (13) and (14), these constraints enforce that task (i) in unit (j) at event point (n+1) starts immediately after the end of task (i') in unit (j') at event point (n) if both of them are activated.

Time Horizon Constraints

$$t^f(i,j,n) \leq H, \quad \forall i \in I, j \in J_i, n \in N \tag{16}$$

$$t^s(i,j,n) \leq H, \quad \forall i \in I, j \in J_i, n \in N \tag{17}$$

The time horizon constraints represent the requirement that every task should start and end within the time horizon (H).

Objective: Minimize

$$\sum_j (\tilde{\alpha}_j e(j) + \tilde{\beta}_j s(j)^{\tilde{\gamma}_j}) - \sum_s \sum_n p_s d(s,n) \tag{18}$$

The objective is to minimize the capital costs of units, which consist of a fixed term and a generally nonlinear term depending on the sizes of the units, minus profits due to product sales. Other performance criteria can also be incorporated.

It should be pointed out that the case where different units share the same tasks can be accommodated in the above formulation by considering each task in each unit as a different task with the same features. We should also note that due to the nonlinear models of processing time and capital cost, the resulting mathematical programming model is a nonconvex MINLP problem. Therefore, deterministic global optimization methods are needed to determine the global optimal solution.

5 Computational Studies

The above mathematical formulation is applied to two examples taken from Xia and Macchietto (1997). For each example, both linear and nonlinear cases are studied. MINOPT, an advanced modeling language and algorithmic framework proposed by Schweiger and Floudas (1997) (URL: http://titan.princeton.edu/MINOPT/), is used to establish and solve the resulting MILP/MINLP mathematical programming problems. The MILP problems are solved using CPLEX, a branch and bound method. For MINLP problems, MINOPT implements several algorithms including Generalized Benders Decomposition (GBD), Outer Approximation with Equality Relaxation (OA/ER), and Outer Approximation with Equality Relaxation and Augmented Penalty (OA/ER/AP). A detailed account of the theoretical and algorithmic issues for MINLP problems can be found in Floudas (1995). All the computations are done on a HP-C160 workstation.

BM Plant

The process recipe, equipment data and plant superstructure are those we discussed in a previous session as BM process and BM plant (see Figure 1, Table 1 and Figure 2, respectively). Further information on the production requirement and relevant material data are shown in Table 2. The time horizon under consideration is 12 hours.

In the case of BMFIX, the processing times of processing tasks and capital cost models are provided as linear forms. Five event points are used and the resulting MILP problem involves 59 binary variables, 175 continuous variables and 332 constraints. It is solved in 0.4 seconds CPU time and the objective function value is 195.6. The optimal plant design is shown in Figure 3, in which all the three processing units are selected while the storage vessel is not. The corresponding optimal operational schedule is represented in Figure 4.

In the case of BMNON, the processing time and capital cost models are both nonlinear. Four event points are needed and the resulting MINLP problem involves 48 binary variables, 142 continuous variables and 262 constraints. It is solved in 0.75 seconds CPU time and the corresponding objective function value is 3557.35. The optimal plant structure is the same as that in BMFIX, as shown in Figure 3. The operational schedule obtained is represented in Figure 5.

KPS Plant

The process recipe is shown in Figure 6. Two final products are produced from three raw materials through heating, three reactions and separation, involving four intermediate materials. The equipment data are given in Table 3. Four processing units and four storage vessels are under consideration. Full connectivity of the equipment network is assumed. The plant superstructure is then constructed as shown in Figure 7. Production requirements and relevant material data are given in Table 4. The time horizon under consideration is also 12 hours.

In the linear case of KPSLIN, six event points are used and the resulting MILP problem involves 128 binary variables, 341 continuous variables and 877 constraints. It is solved in 22.49 seconds CPU time and the objective function value is 572.898. The optimal plant design is shown in Figure 8, in which all four processing units are chosen and only one storage vessel is selected to hold one of the intermediate materials. The corresponding optimal operational schedule is represented in Figure 10.

In the nonlinear case of KPSNON, five event points are needed and the resulting MINLP problem involves 108 binary variables, 287 continuous variables and 722 constraints. It is solved in 7.31 seconds CPU time and the corresponding objective function value is 490.433. Compared to the solution of KPSLIN, only one reactor is included in the optimal plant structure in this case, as shown in Figure 9, which is due to the increased capital cost of the other reactor. The operational schedule obtained is represented in Figure 11.

Comparison with Other Approaches

Table 5 shows the results of the proposed formulation compared with the results found in literature. Xia and Macchietto (1997) transformed the formulation they presented into an alternative one without giving the necessary details of the transformation and based the data they provided on the transformed one. Therefore, in addition to the reported data, the corresponding data we obtain according to their original formulation are also presented here, which in our point of view, reflect the actual size of their resulting mathematical models. It is shown that the formulation proposed in this paper has the following advantages: (i) It gives rise to a simpler mixed-integer optimization problem mainly in terms of a smaller number of binary variables. The proposed formulation introduces 59, 48, 128 and 108 binary variables in BMFIX, BMNON, KPSLIN and KPSNON, respectively, compared to 124 in the first example and 288 in the second example required for the formulation of Xia and Macchietto (1997); (ii) The optimal solution obtained corresponds to a better objective function value and consequently a better integrated design and scheduling strategy. In BMFIX and BMNON, the proposed formulation leads to an objective function value of 195.6 and 3557.35 respectively, which are lower than the value of 197.23 and 3576.49 from the formulation of Xia and Macchietto (1997). In KPSLIN and KPSNON, we also achieve better values of 572.808 and 490.433 compared to 585.62 and 495.11; (iii) The computational effort required is significantly reduced, which makes it very promising to solve large-scale industrial problems. It takes only 0.4 seconds in BMFIX, 0.75 seconds in BMNON, 22.49 seconds in KPSLIN and 7.31 seconds in KPSNON, respectively, on a HP-C160 workstation, while the formulation of Xia and Macchietto (1997) required 1821.19 seconds, 2998.46 seconds, 2407.62 seconds and 7849.23 seconds, respectively, on a Sun Ultra station-1.

Local versus Global Solution

It is interesting to note that in the nonlinear nonconvex cases, BMNON and KPSNON, the solutions obtained through MINOPT (Schweiger and Floudas (1997)), which is a local MINLP solver, are also the global optimal solutions. This is proven by solving the lower-bounding problem, which gives exactly the same solution, that is, the same objective function value, plant structure design and operational schedule. The nonlinearity comes from the variable terms in the processing time model and capital cost model (see Constraints (9) and the objective function (18)). When the exponential constants of these terms, that is, γ_{ij} and $\tilde{\gamma}_j$, are greater than one, these nonlinear terms of continuous variables are convex. We can transform the original nonconvex MINLP problem to a lower-bounding convex problem by splitting the nonlinear equality constraint (9) into two inequality constraints and then relaxing the concave one with underestimating techniques (Floudas (2000); Adjiman et al. (1998a,b)) to a linear constraint as follows:

$$t^f(i,j,n) \leq t^s(i,j,n) + \alpha_{ij} wv(i,n) + \beta_{ij} (V_j^{max})^{\gamma_{ij}-1} b(i,j,n)$$
$$\forall i \in I_p, \quad j \in J_i, \quad n \in N \qquad (19)$$

It is found that only the convex inequality constraint is active at the optimal solution point of the resulting lower-bounding convex problem. In other words, the equality constraint in the original problem can be changed to the corresponding convex inequality constraint without changing the optimal solution, which transforms the mathematical model from a nonconvex problem to a convex one and guarantees automatically the global optimality of the solution obtained. Such fast convergence to the global optimal solution in only one iteration is observed infrequently and it represents an excellent element of the proposed approach. This special property is related to the special forms of the objective function and the constraints, and the bounds of the relevant variables.

6 Conclusions

In this paper, a continuous-time formulation is proposed for integrated design, synthesis and scheduling of multipurpose batch plants on the basis of a previously presented short-time scheduling model. Two computational studies are presented to demonstrate the effectiveness of the proposed formulation. The computational results are compared with those in literature and show that the proposed formulation results in smaller size MILP/MINLP mathematical models primarily in terms of binary variables and better objective values can be accomplished with significantly less computational efforts. An interesting property concerning the global optimality of the solutions obtained in nonconvex nonlinear cases is also discussed.

Acknowledgments

The authors gratefully acknowledge support from the National Science Foundation, the Mobil Technology Company, and Elf-Atochem Company.

References

Adjiman C., Androulakis I., and Floudas C., 1998b, A global optimization method, α bb, for general twice-differentiable constrained NLPs - II. implementation and computational results. *Comp. Chem. Engng.* **22**, 1159–1179.

Adjiman C., Dallwig S., Floudas C., and Neumaier A., 1998a, A global optimization method, α bb, for general twice-differentiable constrained NLPs - I. theoretical advances. *Comp. Chem. Engng.* **22**, 1137–1158.

Barbosa-Póvoa A. and Macchietto S., 1994, Detailed design of multipurpose batch plants. *Comp. Chem. Engng.* **18**, 1013–1042.

Birewar D. and Grossmann I., 1989, Incorporating scheduling in the optimal design of multiproduct batch plants. *Comp. Chem. Engng.* **13**, 114–161.

Floudas C., 1995, *Nonlinear and Mixed-Integer Optimization*. Oxford University Press.

Floudas C., 2000, *Deterministic Global Optimization: Theory, Methods and Applications*. Kluwer Academic Publishers.

Grossmann I. and Sargent R., 1979, Optimal design of multipurpose chemical plants. *Ind. Eng. Chem. Process Des. Dev.* **18**, 343–348.

Ierapetritou M.G. and Floudas C.A., 1998a, Effective continuous-time formulation for short-term scheduling: 1. multipurpose batch processes. *Ind. Eng. Chem. Res.* **37**, 4341–4359.

Ierapetritou M.G. and Floudas C.A., 1998b, Effective continuous-time formulation for short-term scheduling: 2. continuous and semi-continuous processes. *Ind. Eng. Chem. Res.* **37**, 4360–4374.

Ierapetritou M.G., Hené T.S., and Floudas C.A., 1999, Effective continuous-time formulation for short-term scheduling: 3. multiple intermediate due dates. *Ind. Eng. Chem. Res.* **38**, 3446–3461.

Kondili E., Pantelides C., and Sargent R., 1993, A general algorithm for short-term scheduling of batch operations - I. MILP formulation. *Comp. Chem. Engng.* **17**, 211–227.

Papageorgaki S. and Reklaitis G., 1990a, Optimal design of multipurpose batch processes.1.problem formulation. *Ind. Eng. Chem. Res.* **29**, 2054–2062.

Papageorgaki S. and Reklaitis G., 1990b, Optimal design of multipurpose batch processes.2. a decomposition solution strategy. *Ind. Eng. Chem. Res.* **29**, 2062–2073.

Papageorgaki S. and Reklaitis G., 1993, Retrofitting a general multipurpose batch chemical plants. *Ind. Eng. Chem. Res.* **32**, 345–362.

Schweiger C. and Floudas C., 1997, *MINOPT : A Software Package for Mixed–Integer Nonlinear Optimization, User's Guide*. Computer–Aided Systems Laboratory, Dept. of Chemical Engineering, Princeton University, NJ.

Suhami I. and Mah R., 1982, Optimal design of multipurpose batch plants. *Ind. Eng. Chem. Process Des. Dev.* **21**, 94–100.

Xia Q. and Macchietto S., 1997, Design and synthesis of batch plants - MINLP solution based on a stochastic method. *Comp. Chem. Engng.* **21**, S697–S702.

Zhang X. and Sargent R., 1996, The optimal operation of mixed production facilities - general formulation and some solution approaches for the solution. *Comp. Chem. Engng.* **20**, 897–904.

Zhang X. and Sargent R., 1998, The optimal operation of mixed production facilities - extensions and improvements. *Comp. Chem. Engng.* **20**, 1287–1295.

List of Figures

List of Tables

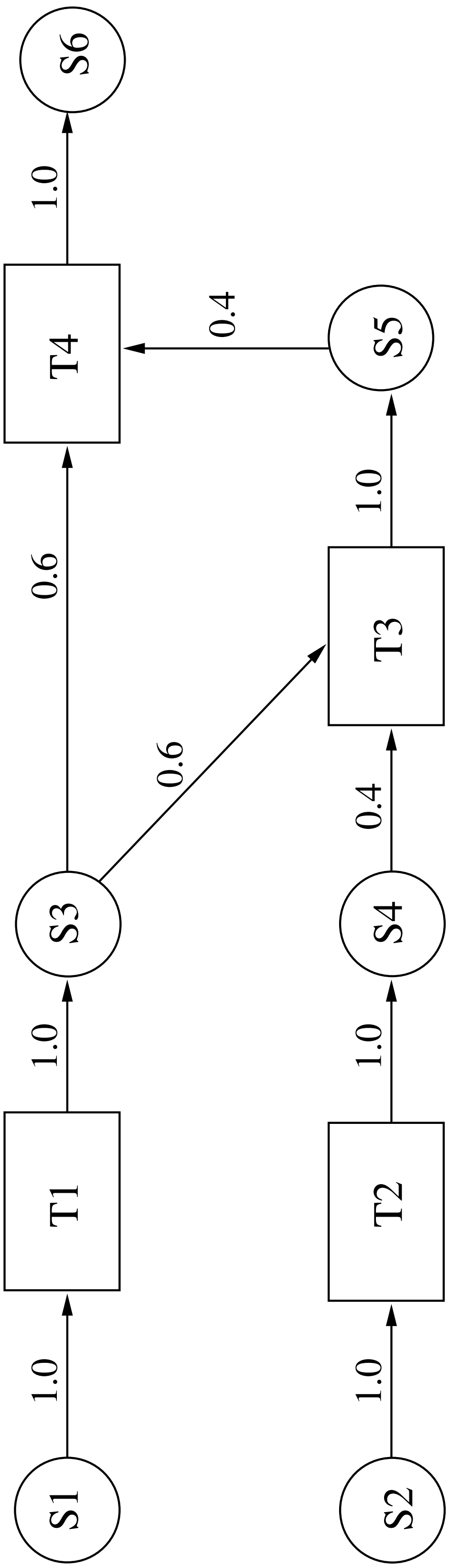

Figure 1: State-Task Network Representation for BM Process

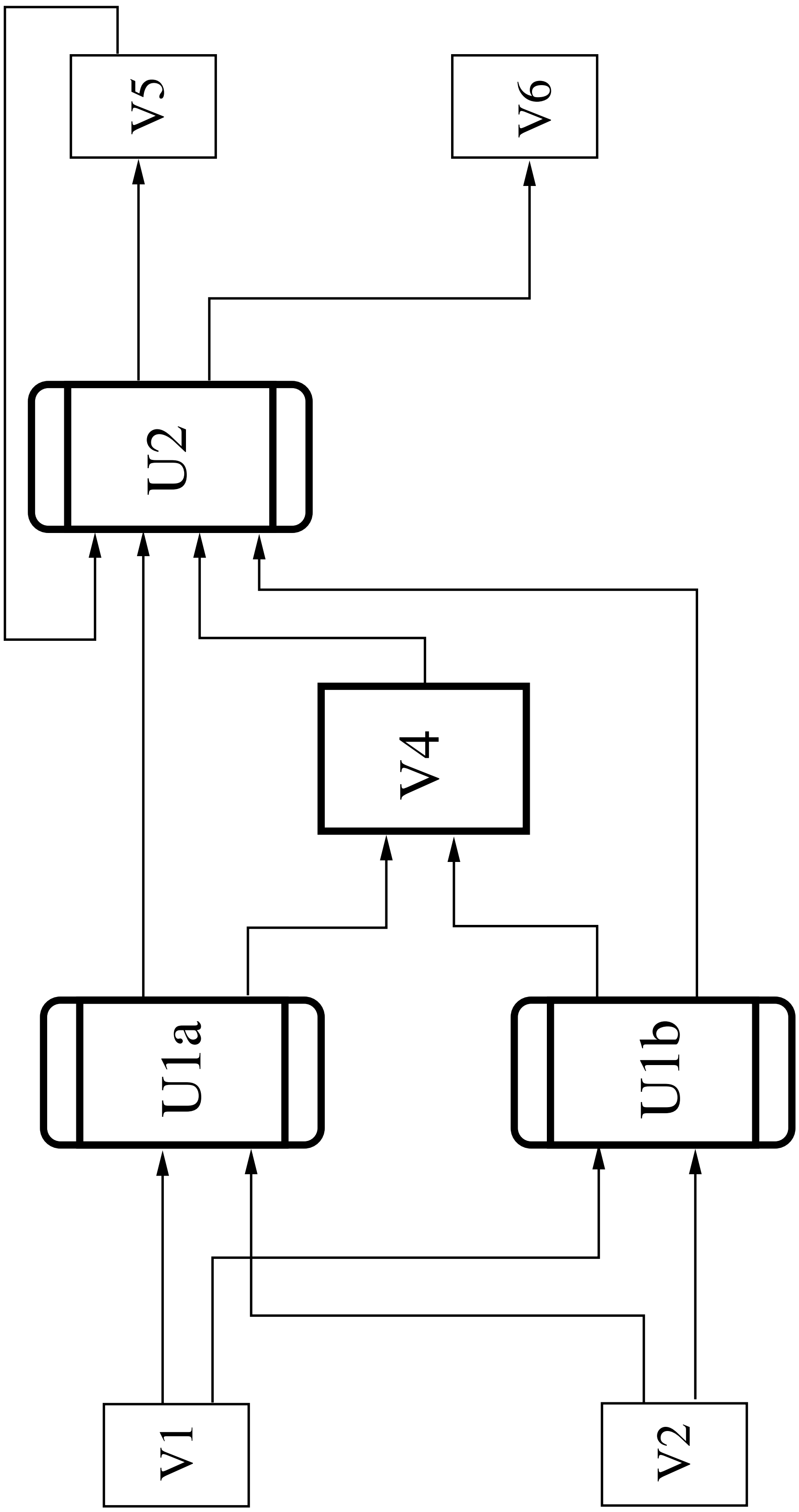

Figure 2: Plant Superstructure for BM Plant

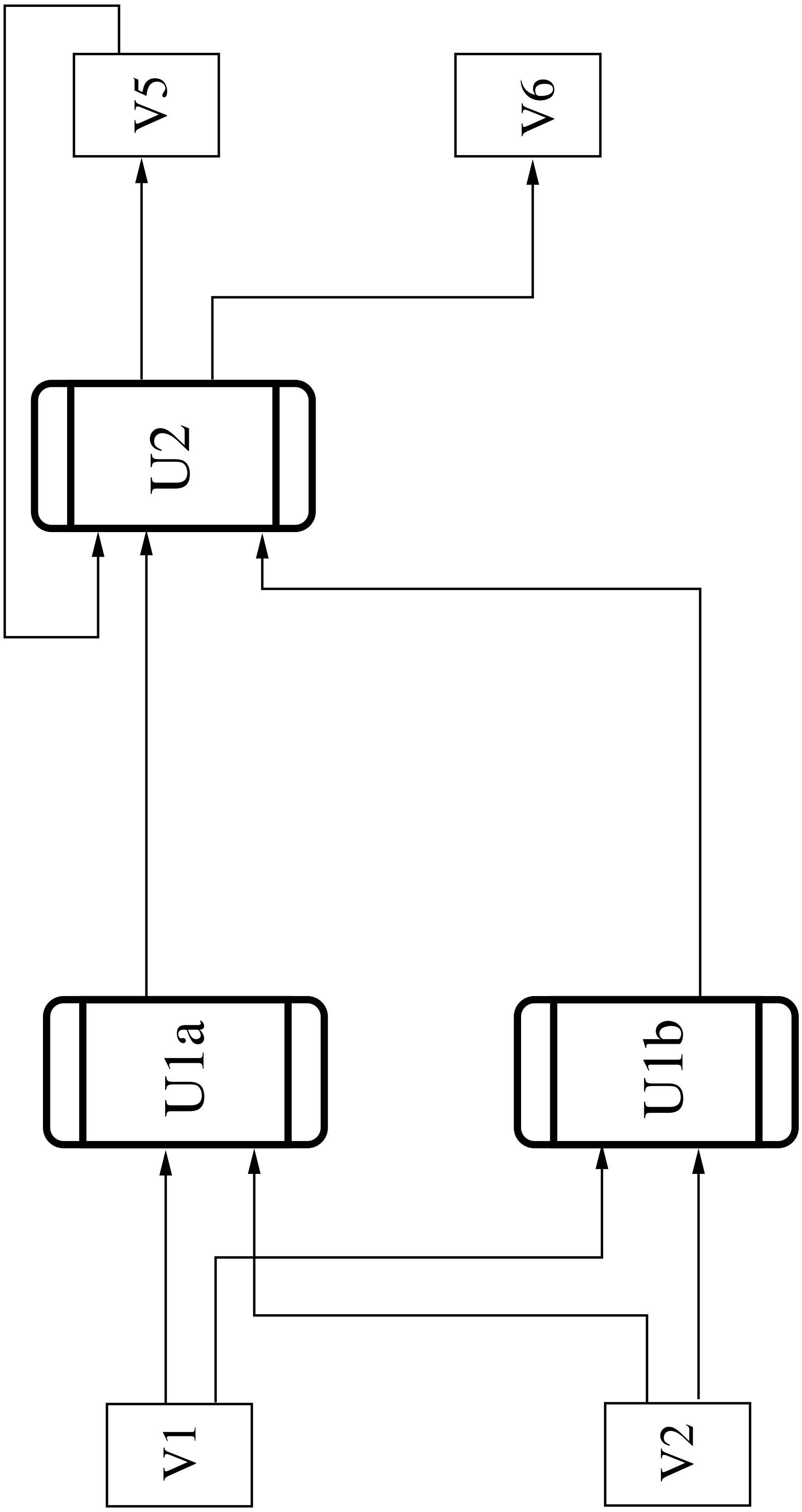

Figure 3: Optimal Plant Structure for BMFIX and BMNON

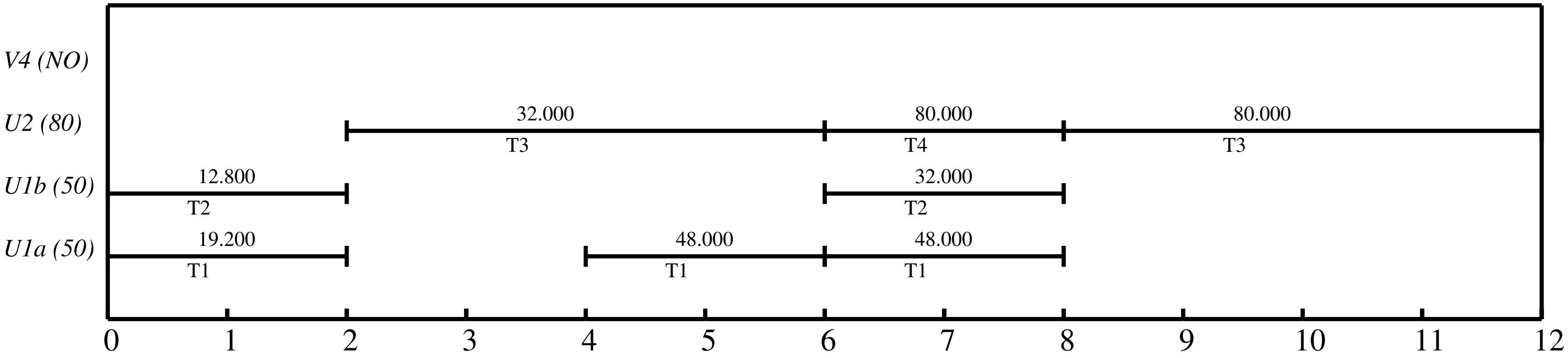

Figure 4: Gantt Chart for BMFIX

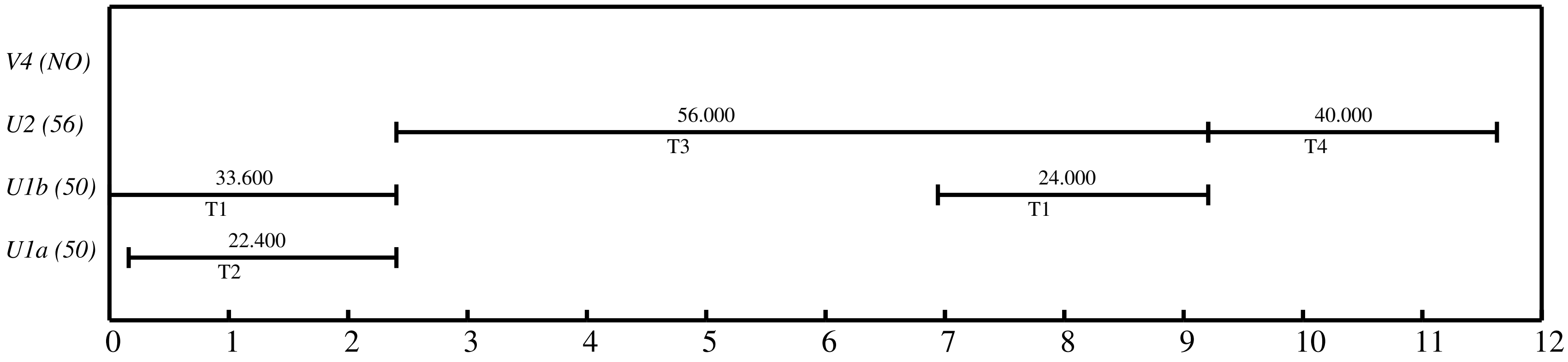

Figure 5: Gantt Chart for BMNON

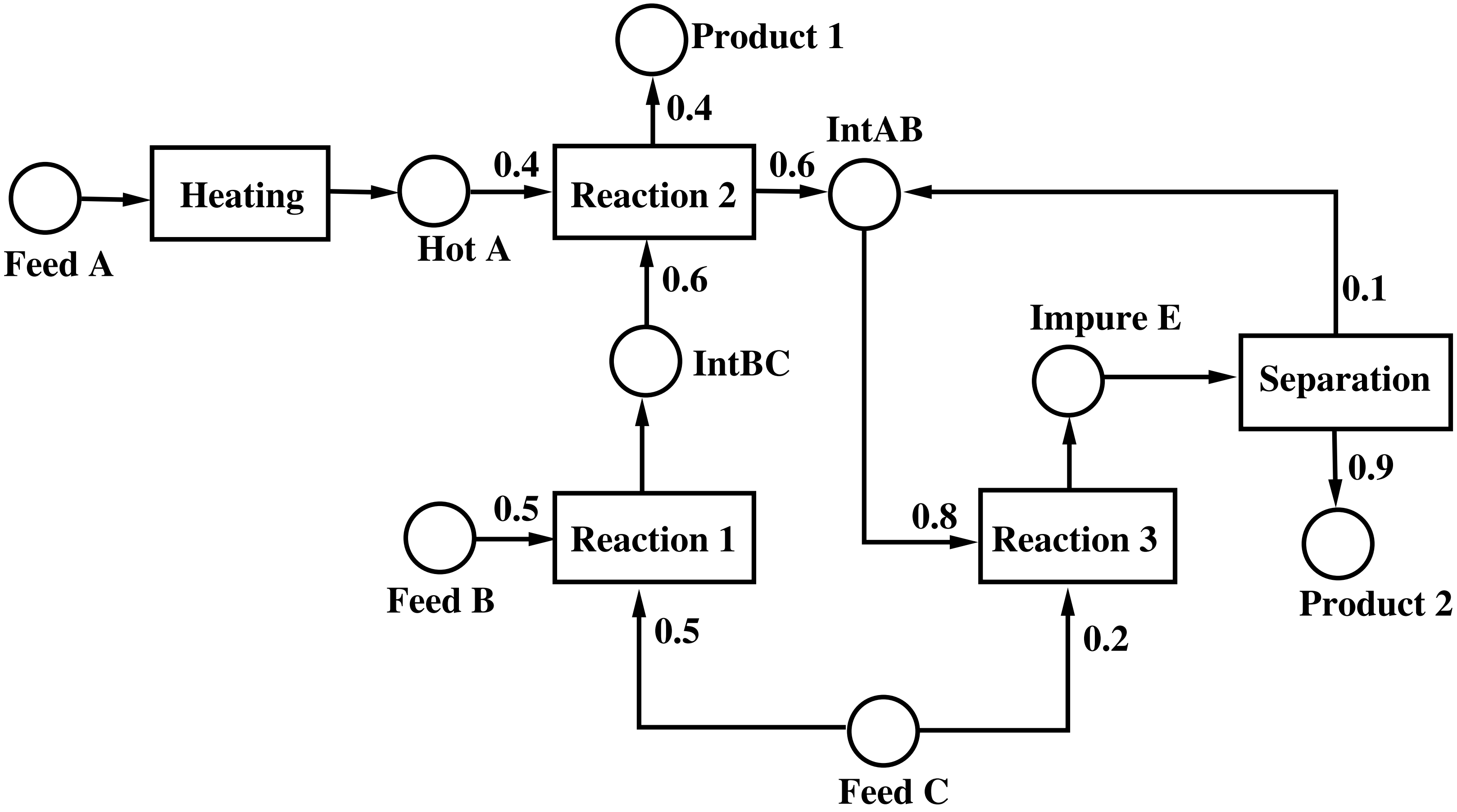

Figure 6: State-Task Network Representation for KPS Process

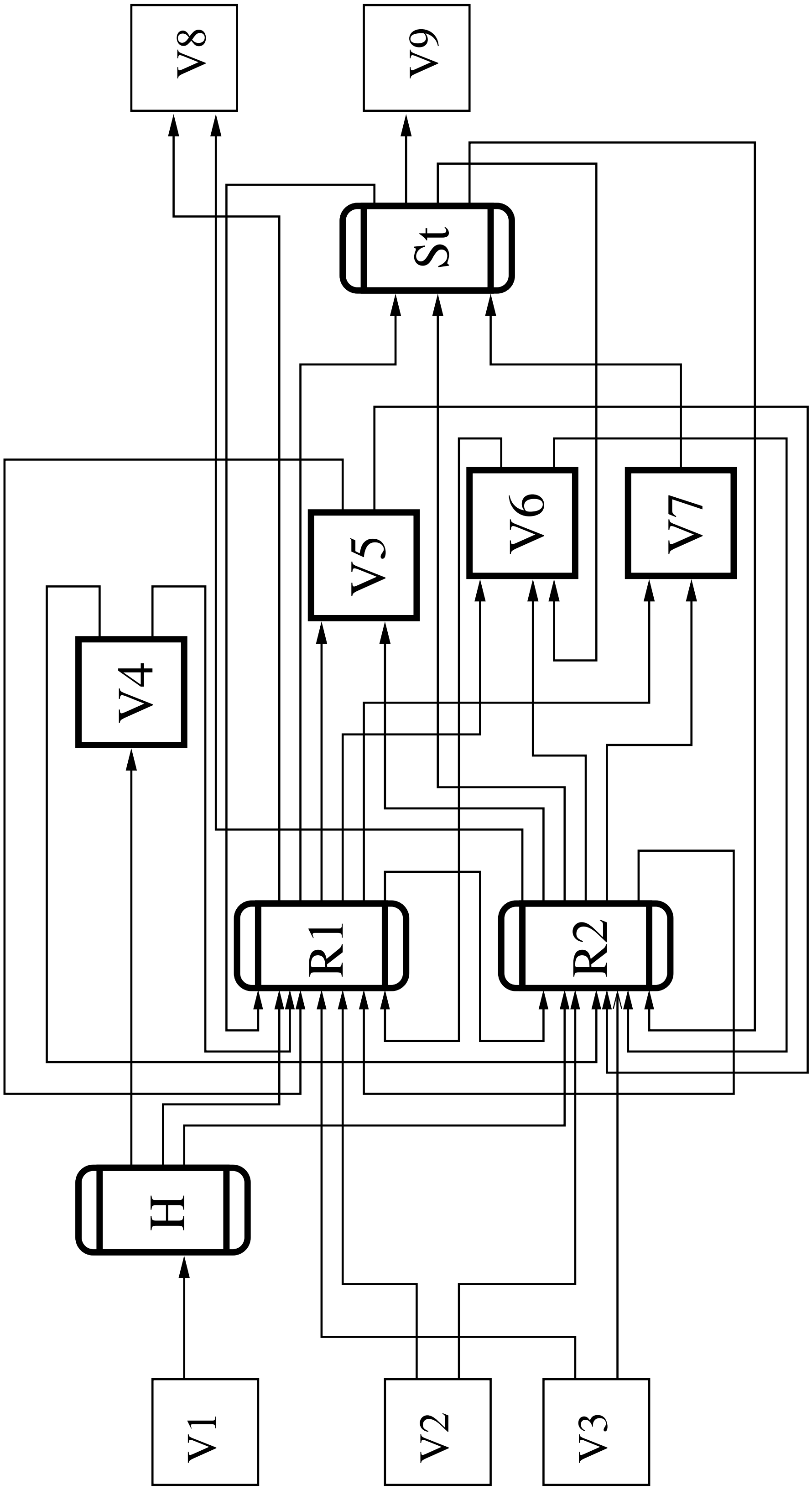

Figure 7: Plant Superstructure for KPS Plant

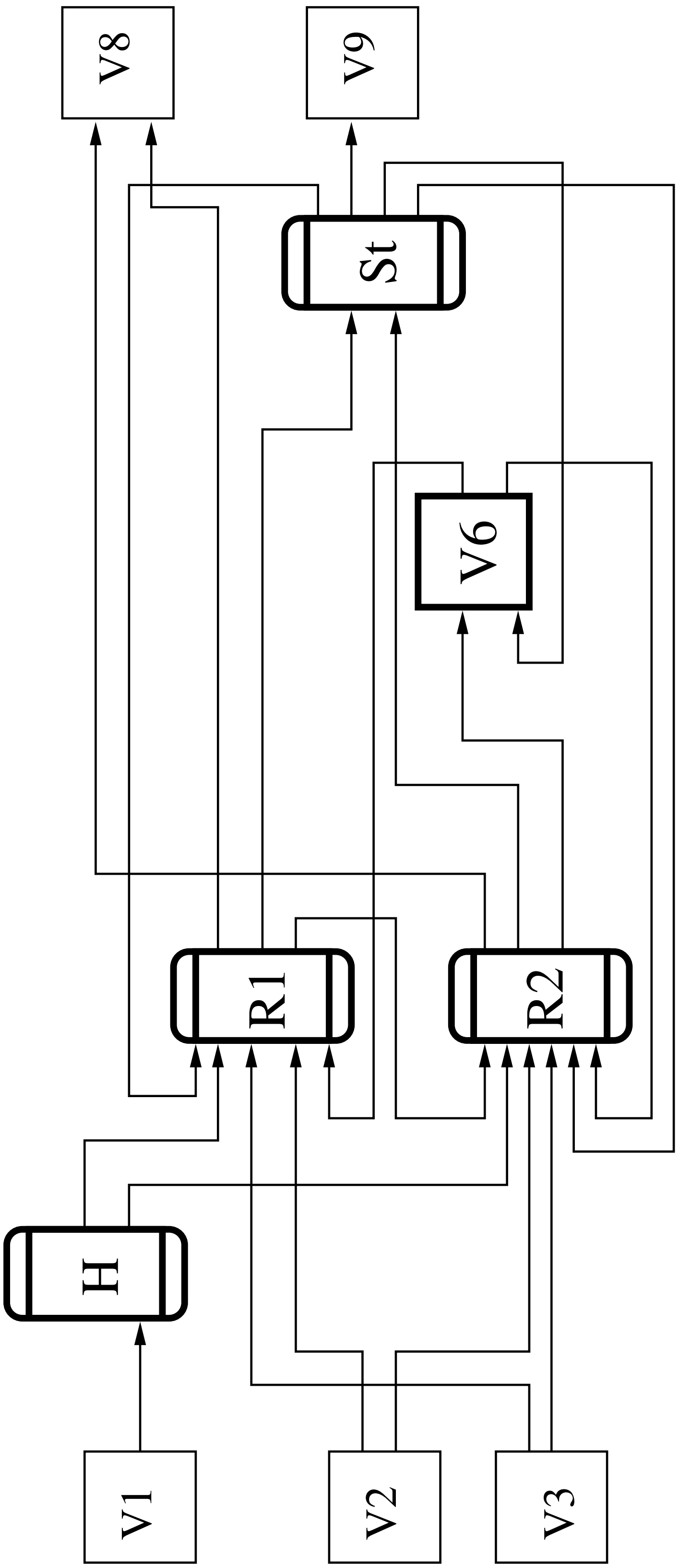

Figure 8: Optimal Plant Structure for KPSLIN

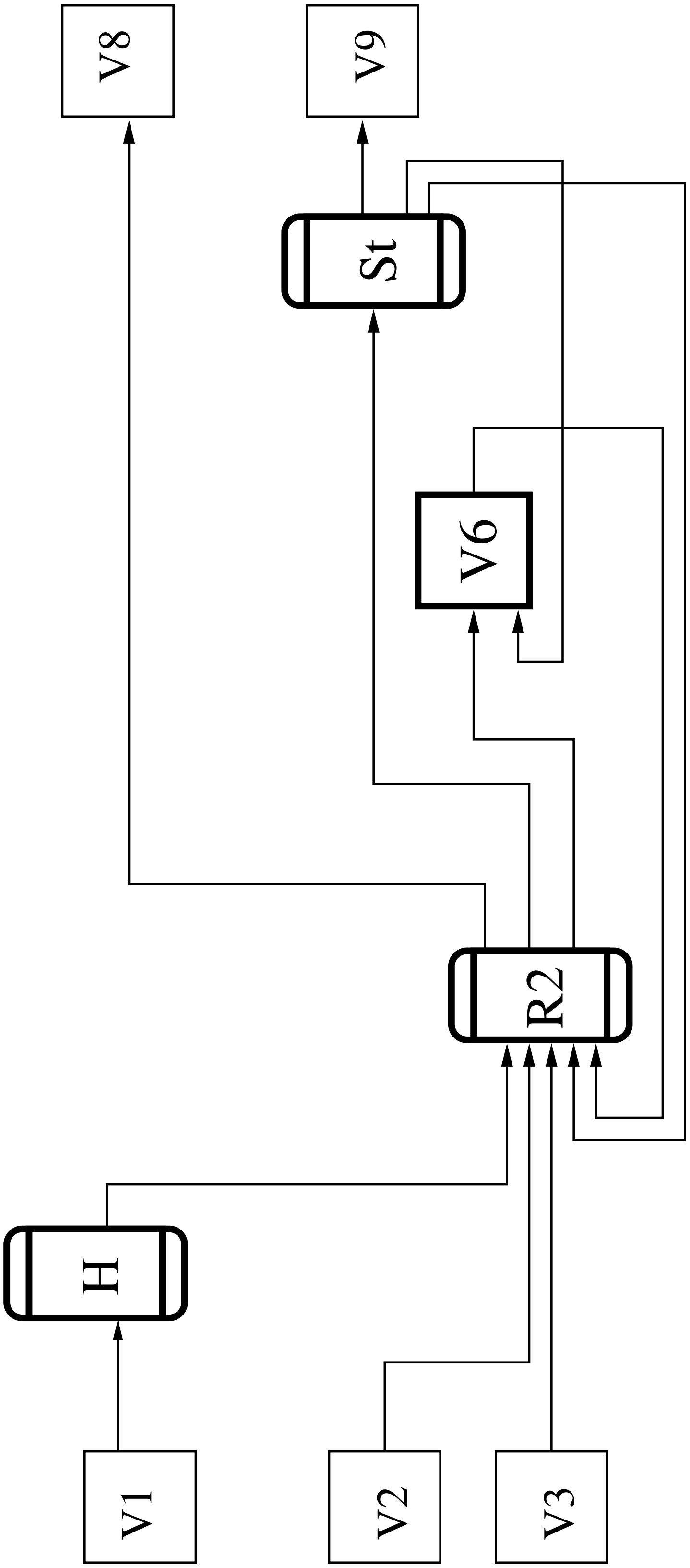

Figure 9: Optimal Plant Structure for KPSNON

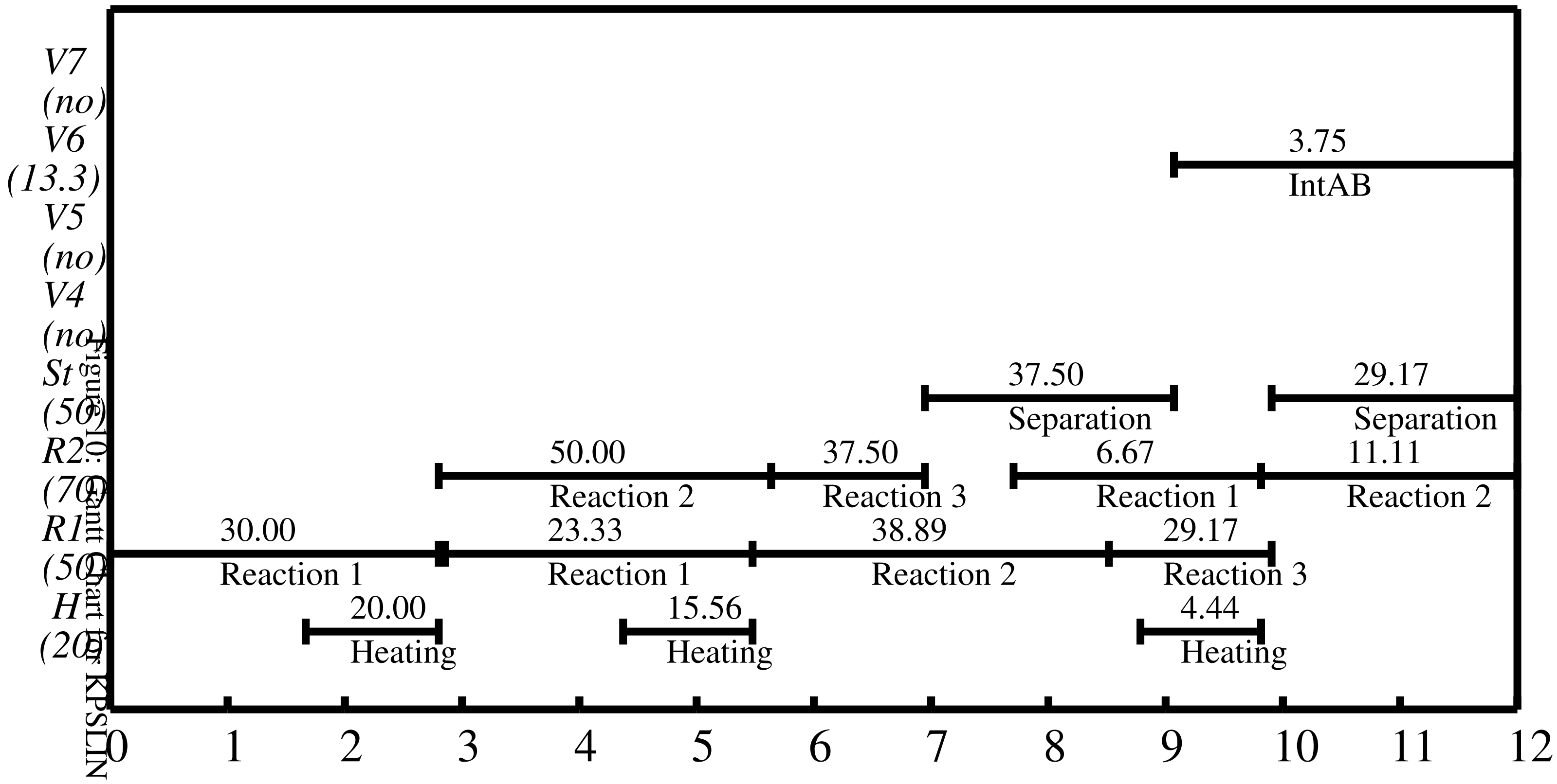

Figure 10: Gantt Chart for KPSLIN

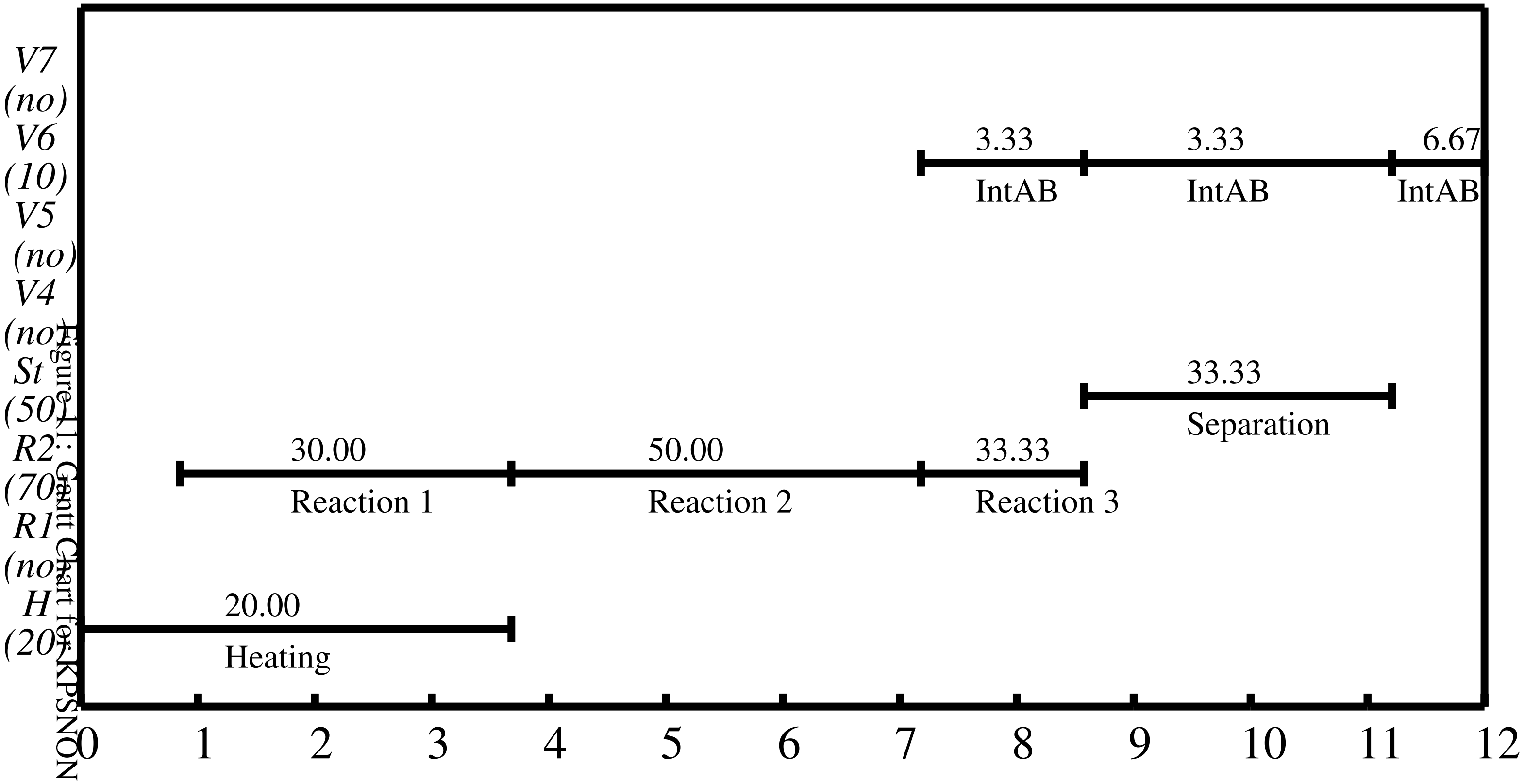

Figure 11: Gantt Chart for KPSNON

Unit	Capacity	Suitability	Task Time Model		Cost Model	
			BMFIX	BMNON	BMFIX	BMNON
Units 1a,1b	50-150	T1, T2	2.0	$2.0 + 0.005b^{1.25}$	$20 + 0.5s$	$20.0 + 0.5s^{1.5}$
Unit 2	50-200	T3	4.0	$4.0 + 0.01b^{1.4}$	$30 + 1.0s$	$30 + 1.0s^{2.0}$
		T4	2.0	$2.0 + 0.005b^{1.2}$		
Vessel 4	10-100	S4	–	–	$1.0 + 0.1s$	$1.0 + 0.1s^{1.75}$

Table 1: Equipment Data for BM Plant

State	Storage Capacity	Price	Requirement	
			BMFIX	BMNON
S1	Unlimited	0	0	0
S2	Unlimited	0	0	0
S3	0	0	0	0
S4	Vessel 4	0	0	0
S5	Unlimited	0.04	80.0	40.0
S6	Unlimited	0.015	80.0	40.0

Table 2: Material Data for BM Plant

Unit	Capacity	Suitability	Task Time Model		Cost Model	
			KPSLIN	KPSNON	KPSLIN	KPSNON
Heater	20-50	Heating	$1.0 + 0.0067b$	$1.0 + 0.0067b^{2.00}$	$100.0 + 0.2s$	$100.0 + 0.2s$
Reactor 1	50-70	Reaction 1	$2.0 + 0.0267b$	$2.0 + 0.0267b^{1.25}$	$150.0 + 0.5s$	$150.0 + 0.5s^{1.5}$
		Reaction 2	$2.0 + 0.0267b$	$2.0 + 0.0267b^{1.25}$		
		Reaction 3	$1.0 + 0.0133b$	$1.0 + 0.0133b^{1.05}$		
Reactor 2	70	Reaction 1	$2.0 + 0.0167b$	$2.0 + 0.0167b^{1.15}$	120.0	120.0
		Reaction 2	$2.0 + 0.0167b$	$2.0 + 0.0167b^{1.15}$		
		Reaction 3	$1.0 + 0.0083b$	$1.0 + 0.0083b^{1.10}$		
Still	50-80	Separation	$2.0 + 0.0033b$	$2.0 + 0.0033b^{1.50}$	$150.0 + 0.3s$	$150.0 + 0.3s^{1.5}$
Vessel 4	10-30	I1(Hot A)	–	–	$30.0 + 0.1s$	$30.0 + 0.1s$
Vessel 5	10-60	I2(IntBC)	–	–	$15.0 + 0.1s$	$15.0 + 0.1s^{1.5}$
Vessel 6	10-70	I3(IntAB)	–	–	$10.0 + 0.1s$	$10.0 + 0.1s^{1.2}$
Vessel 7	50-100	I4(Impure E)	–	–	$20.0 + 0.2s$	$20.0 + 0.2s^{1.5}$

Table 3: Equipment Data for KPS Plant

State	Storage Capacity	Price	Requirement	
			KPSLIN	KPSNON
Feed A	Unlimited	-0.001	0	0
Feed B	Unlimited	-0.002	0	0
Feed C	Unlimited	-0.0015	0	0
Hot A	Vessel 4	0	0	0
IntBC	Vessel 5	0	0	0
IntAB	Vessel 6	0	0	0
Impure E	Vessel 7	0	0	0
Product 1	Unlimited	0.02	40.0	20.0
Product 2	Unlimited	0.03	60.0	30.0

Table 4: Material Data for KPS Plant

Case	Formulation	Cost ($\$10^3$)	Integer Variables	Continuous Variable	Constraints	CPU (sec)
BMFIX	Xia and Macchietto(1997)	197.23	40^t	22^t	55^t	1821.19^*
			124^o	107^o	220^o	
	This Work	195.6	59	175	332	0.4^{**}
BMNON	Xia and Macchietto(1997)	3576.49	40^t	22^t	55^t	2998.46^*
			124^o	107^o	220^o	
	This Work	3557.35	48	142	262	0.75^{**}
KPSLIN	Xia and Macchietto(1997)	585.62	62^t	34^t	122^t	2407.62^*
			288^o	201^o	425^o	
	This Work	572.898	128	341	877	22.49^{**}
KPSNON	Xia and Macchietto(1997)	495.11	62^t	34^t	122^t	7849.23^*
			288^o	201^o	425^o	
	This Work	490.433	108	287	722	7.31^{**}

Table 5: Results and Comparisons (t: reported based on transformed formulation; o: recounted based on original formulation; *: Sun Ultra station-1 ; **: HP-C160 workstation)

Abstract

Design, synthesis and scheduling issues are considered simultaneously for multipurpose batch plants. A previously proposed continuous-time formulation for scheduling is extended to incorporate design and synthesis. Processing recipes are represented by the State-Task Network. The superstructure of all possible plant designs is constructed according to the potential availability of all processing/storage units. The proposed model takes into account the trade-offs between capital costs, revenues and operational flexibility. Computational studies are presented to illustrate the effectiveness of the proposed formulation. Both linear and nonlinear models are included, resulting in MILP and MINLP problems, respectively. The MILP problems are solved using a branch and bound method. Globally optimal solutions are obtained for the nonconvex MINLP problems based on a key property that arises due to the special structure of the involved problems. Comparisons with another approach are also presented.