# Effective Continuous-Time Formulation for Short-Term Scheduling: II: Continuous and Semi-continuous Processes

M.G. Ierapetritou and C. A. Floudas[1]
Department of Chemical Engineering,
Princeton University,
Princeton, NJ 08544-5263, USA

**Abstract**– Part I presented a novel continuous-time mathematical formulation for the short-term scheduling of batch processes. Most production networks however, involve batch and continuous processes. Based on the same principles, this paper extends the proposed formulation to describe continuous processes. Two industrial case studies from the Fast Moving Consumer Goods Manufacturing are presented to illustrate the capability of the proposed formulation to describe plants with both batch and continuous processes, to incorporate clean-up requirements and to consider storage requirements and limitations. It is demonstrated that the proposed approach outperforms all previously proposed continuous-time models for the short-term scheduling of continuous processes.

## 1 Introduction

A significant body of research has appeared in the Chemical Engineering literature concerning the short-term scheduling of batch processes, while the consideration of short term scheduling of continuous processes has received less attention even though industrial plants usually involve both batch and continuous processes. Moreover, the increasing need for moving towards flexible multiproduct continuous plants that respond to market requirements quickly, provides an additional motivation for investigating more closely the scheduling of continuous processes.

Recent publications in the area of scheduling of continuous plants involve the work of Sahinidis and Grossmann[1] that examined the long-term planning for the continuous production of multiple lines. They formulated an MINLP problem to determine the optimal solution based on cyclic operation. Pinto and Grossmann[2] considered the problem of multiple stages with intermediate storage considerations also based on cyclic operation patterns for long term horizons and they modeled the problem as an MINLP problem for which they use decomposition based solution techniques. Kondili et al.[3] studied the problem of planning a multiproduct energy-intensive continuous cement milling plant. They utilized a hybrid discrete-continuous time formulation considering discrete time periods and time slots of varying duration. The resulting MILP problem was solved with conventional MILP solvers based on branch and bound (B&B) principles.

[1]Author to whom all correspondence should be addressed; Tel: (609) 258-4595; Fax:(609) 258-0211.

Considering the continuous nature of the processes, it becomes even more apparent that discrete time representation becomes much less appropriate for describing the scheduling problem for the continuous processes. Continuous time formulations for the short-term scheduling problem have been developed by Zhang and Sargent[4] and Schilling and Pantelides[5]. Zhang and Sargent[4] used the RTN representation of the plant flowsheet to formulate a large-scale MILP formulation. Schilling and Pantelides[5] also utilized the RTN representation for their continuous time formulation for batch and continuous processes. They proposed a special B&B solution procedure branching on binary and continuous variables in order to improve the computational efficiency of the proposed formulation. McDonald and Karimi[6;7] proposed mathematical models for production planning and short-term scheduling. For the problem of short-term scheduling they proposed two mathematical models that differ on the preassignment of slots to time periods. The proposed formulations can handle the problem of a single-stage multiproduct facility with parallel semi-continuous processors. The model complexity requires the use of preassignment slots to time periods, as well as problem decomposition in order to address medium-size problems.

In this paper, the formulation proposed in Part I is extended to accommodate the description of continuous processes. In particular, in the next section the mathematical model and its basic characteristics are presented. Two case studies from a Fast Moving Consumer Manufacturing are then presented in section 3 to illustrate the capability of the proposed approach to large scale problems. These case studies are also used to illustrate the incorporation of clean-up requirements within the proposed framework, as well as the consideration of storage limitations. Section 3 presents the results of the proposed formulation and comparisons with the existing continuous time formulations.

# 2 Mathematical Formulation

The proposed formulation is based on the same fundamental components of the framework presented in part I of the paper for the short-term scheduling of batch processes. In particular, it follows a *continuous time* representation, it uses different binary variables for *task events* and *unit events* and allows for variable processing times with respect to the amount of material processed by the specific task. The proposed formulation utilizes the State Task Network representation[8] as a general way of describing the production network. Based on these ideas the following formulation is developed where the indices, sets, parameters and variables are as follows:

*Indices:*

- i tasks;
- j units;
- n event points representing the beginning of a task;
- s states.

*Sets:*

- $I$ tasks;
- $I_j$ tasks which can be performed in unit (j);

$I_s$ tasks which process state (s) and either produce or consume;

$I_{st}$ storage tasks;

$J$ units;

$J_i$ units which are suitable for performing task (i);

$J_{st}$ storage units;

$N$ event points within the time horizon;

$S$ set of all involved states (s);

$S^R$ subset of S that involves the raw materials;

$S^P$ subset of S that involves the products;

$S^{IN}$ subset of S that involves the intermediates.

*Parameters:*

$R_{ij}^{min}$ denotes the minimum rate of material processed by task (i) required to start operating unit (j);

$R_{ij}^{max}$ denotes the maximum rate of the specific unit (j) when processing task (i);

$R_{ij}$ denotes the production rate of task (i) processed in unit (j) in cases that these are constant ;

$V^{max}(i_{st}, j_{st})$ available storage capacity of storage unit $(j_{st})$ when processing storage task $(i_{st})$;

$\rho_{si}^p, \rho_{si}^c$ proportion of state (s) produced, consumed from task (i), respectively, $\rho_{si}^p \geq 0, \rho_{si}^c \leq 0$

$d^{req}(s)$ demand of state (s) at the end of the time horizon as denoted by the market requirements;

H time horizon;

shift time duration of the shift;

$t_{jii'}^{cl}$ clean-up time required between tasks (i) and $(i')$ at unit (j);

price(s) price of state (s).

*Variables:*

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

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

B(i,j,n) amount of material undertaking task (i) in unit (j) at point (n);

$d(s,n)$ amount of state (s) being delivered to the market at point (n);

$ST_0(s,n)$ amount of state (s) $\in S^R$ that is available as required from external recourses at any event point (n);

$SB(s,n)$ amount of state (s) $\in S^{IN}$ that goes directly to the production task bypassing storage at the event point (n);

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

$T^f(i,j,n)$ time that task (i) finishes in unit (j) at point (n).

Based on this notation the mathematical model for short term scheduling of continuous plants involves the following constraints:

**Allocation Constraints**

$$\sum_{i \in I_j} wv(i,n) = yv(j,n), \quad \forall j \in J, n \in N \tag{1}$$

These constraints express that if a task (i) starts at point (n) it should take place in one of the suitable units ($j \in J_i$).

**Capacity Constraints**

$$R_{ij}^{min}[T^f(i,j,n) - T^s(i,j,n)] \leq B(i,j,n) \leq R_{ij}^{max}[T^f(i,j,n) - T^s(i,j,n)], \quad \forall i \in I, j \in J_i, n \in N \tag{2}$$

These constraints express the limitations of minimum and maximum rate of unit (j) when performing task (i). The amount of material being processed $B(i,j,n)$ should be between the limits $R_{ij}^{min}[T^f(i,j,n) - T^s(i,j,n)]$ and $R_{ij}^{max}[T^f(i,j,n) - T^s(i,j,n)]$ where $[T^f(i,j,n) - T^s(i,j,n)]$ is the duration of the task (i) in unit (j) at event point (n). If the production rate is constant, then the following equality holds:

$$B(i,j,n) = R_{ij}[T^f(i,j,n) - T^s(i,j,n)] \quad \forall i \in I, j \in J_i, n \in N \tag{3}$$

**Storage Constraints**

$$B(i_{st}, j_{st}, n) \leq V^{max}(i_{st}, j_{st})\ wv(i_{st}, n), \quad \forall i_{st} \in I_{st}, j \in J_{st}, n \in N \tag{4}$$

These constraints represent the maximum available storage capacity for each storage unit ($j_{st}$).

**Material Balances**

Raw Materials:

$$ST_0(s,n) \quad + \quad \sum_{i \in I_s} \rho_{si}^c \sum_{j \in J_i} B(i,j,n) = 0, \quad \forall s \in S^R, n \in N \tag{5}$$

These constraints represent the requirement that the amount of raw material (s) available from the market at point (n) is consumed within the plan at point (n). Note that the above constraints are based on the assumption that the material is available as required from external recourses. The case where the raw materials should be received at the beginning of the scheduling time could be easily accommodated through the following modification of the above constraint that now holds for the first event point:

$$ST_0(s) \quad + \quad \sum_{i \in I_s} \rho^c_{si} \sum_{j \in J_i} B(i,j,0) = 0, \quad \forall s \in S^R \tag{6}$$

where set $I_s$ consists of both production and storage tasks for the raw materials. For the consecutive event points the following balances should hold:

$$\sum_{i \in I_{st}} \rho^p_{si_{st}} \sum_{j_{st} \in J_{i_{st}}} B(i_{st}, j_{st}, n-1) + \sum_{i \in I_s \setminus I_{st}} \rho^c_{si} \sum_{j \in J_i} B(i,j,n) = 0, \quad \forall s \in S^R, n \in N \tag{7}$$

These constraints represent the requirement that the amount of raw material (s) stored at the previous event point should be consumed and the rest should remain at the storage to be used later within the time horizon.

<u>Intermediates:</u>

For the intermediates there exist two sets of constraints, (a) the ones prior to the storage tasks and (b) the ones following the storage tasks:

(a) Prior to the storage tasks:

$$\sum_{i \in I_s \setminus I_{st}} \rho^p_{si} \sum_{j \in J_i} B(i,j,n) + \sum_{i \in I_{st}} \rho^c_{si_{st}} \sum_{j_{st} \in J_{i_{st}}} B(i_{st}, j_{st}, n) - SB(s,n) = 0, \quad \forall s \in S^{IN}, n \in N \tag{8}$$

where $SB(s,n)$ denotes the amount of the intermediate material (s) that goes directly to the production task bypassing storage at the event point (n). According to these constraints, state (s) that corresponds to an intermediate is consumed as soon as it is produced at event point (n) by the corresponding storage tasks or directly by the production tasks.

(b) Following the storage tasks:

$$SB(s,n) + \sum_{i \in I_{st}} \rho^p_{si_{st}} \sum_{j_{st} \in J_{i_{st}}} B(i_{st}, j_{st}, n-1) + \sum_{i \in I_s \setminus I_{st}} \rho^c_{si} \sum_{j \in J_i} B(i,j,n) = 0, \quad \forall s \in S^{IN}, n \in N \tag{9}$$

According to these constraints, state (s) that is just been produced at the event point (n) or has been stored at the previous event point (n-1) is consumed by the production tasks at the event point (n). Note that the variables $SB(s,n)$ can be substituted in (9) using the constraints (8) leading to the following single set of constraints:

$$\sum_{i \in I_s \setminus I_{st}} \rho^p_{si} \sum_{j \in J_i} B(i,j,n) + \sum_{i \in I_{st}} \rho^p_{si_{st}} \sum_{j_{st} \in J_{i_{st}}} B(i_{st}, j_{st}, n-1) +$$
$$\sum_{i \in I_{st}} \rho^c_{si_{st}} \sum_{j_{st} \in J_{i_{st}}} B(i_{st}, j_{st}, n) + \sum_{i \in I_s \setminus I_{st}} \rho^c_{si} \sum_{j \in J_i} B(i,j,n) = 0,$$
$$\forall s \in S^{IN}, n \in N \qquad (10)$$

In the particular case where the production tasks (e.g., mixing) must be followed by the storage tasks first and then the consumption tasks (e.g., packing) as it is in the case study 2, the material balance constraints for the intermediates are split into the stages of (a) production - storage and (b) storage - consumption as follows:
(a) production - storage:

$$\sum_{i \in I_s \setminus I_{st}} \rho^p_{si} \sum_{j \in J_i} B(i,j,n) + \sum_{i \in I_{st}} \rho^c_{si_{st}} \sum_{j_{st} \in J_{i_{st}}} B(i_{st}, j_{st}, n) = 0,$$
$$\forall s \in S^{IN}, n \in N \qquad (11)$$

expressing the requirement that of the intermediate (s) produced at the event point (n) (e.g., via mixing) must be stored at the same event point, and
(b) storage - consumption:

$$\sum_{i \in I_{st}} \rho^p_{si_{st}} \sum_{j_{st} \in J_{i_{st}}} B(i_{st}, j_{st}, n) + \sum_{i \in I_s \setminus I_{st}} \rho^c_{si} \sum_{j \in J_i} B(i,j,n) = 0,$$
$$\forall s \in S^{IN}, n \in N \qquad (12)$$

expressing the requirement that of that intermediate (s) that is stored at the event point (n) is to start undertaking task $i \in I_s \setminus I_{st}$ (e.g., packing) immediately after the storage task begins.

In the case where the intermediate is produced from a batch task, then constraint (8) should be written in the following form to account for the fact that the batch task should first finish in order for the produced material to become available:

$$\sum_{i \in I_s \setminus I_{st}} \rho^p_{si} \sum_{j \in J_i} B(i,j,n-1) + \sum_{i \in I_{st}} \rho^c_{si_{st}} \sum_{j_{st} \in J_{i_{st}}} B(i_{st}, j_{st}, n) - SB(s,n) = 0,$$
$$\forall s \in S^{IN}, n \in N \qquad (13)$$

Products:

$$\sum_{i \in I_s} \rho^p_{si} \sum_{j \in J_i} B(i,j,n) \quad = \quad d(s,n), \quad \forall s \in S^P, n \in N \qquad (14)$$

According to these constraints, the amount of product (s) being produced at point (n) equals the market requirements at this point.
In the case where the product is produced from a batch task, constraint (14) should written in the following form:

$$\sum_{i \in I_s} \rho_{si}^p \sum_{j \in J_i} B(i,j,n-1) \quad = \quad d(s,n), \quad \forall s \in S^P, n \in N \tag{15}$$

**Demand Constraints**

$$\sum_{n \in N} d(s,n) \geq d^{req}(s), \quad \forall s \in S \tag{16}$$

These constraints represent the requirement of production to produce at least as required by the market during the time horizon. Notice, that the requirement of meeting exactly the product demand can be easily accommodated by considering these constraints as equalities.

**Duration Constraints**

$$0.0 \leq T^f(i,j,n) - T^s(i,j,n) \leq (shift)wv(i,n), \quad \forall i \in I, j \in J_i, n \in N \tag{17}$$

expressing the requirement for the task duration to be limited by the duration of production shift and to be positive if this task is processed. Note, that different upper bounds can be used for the duration of the tasks in order to accommodate the specific characteristics of the plant considered.

**Sequence Constraints: Same task in the same unit**

$$\begin{aligned} T^s(i,j,n+1) \geq T^f(i,j,n) - H(2 - wv(i,n) - wv(i,n+1)) \\ \forall i \in I, j \in J_i, n \in N, n \neq N \end{aligned} \tag{18}$$

$$T^s(i,j,n+1) \geq T^s(i,j,n) \;\; \forall i \in I, j \in J_i, n \in N, n \neq N \tag{19}$$

$$T^f(i,j,n+1) \geq T^f(i,j,n) \;\; \forall i \in I, j \in J_i, n \in N, n \neq N \tag{20}$$

expressing the constraint of task (i) starting at point (n+1) to start after the end of the same task performing at the same unit (j) which has already started at point (n). If task (i) takes place at unit (j) at event points (n) and (n+1) in which case $wv(i,n+1) = 1$, and $wv(i,n) = 1$, constraint (18) enforces the starting time of the task (i) at point (n+1), $(T^s(i,j,n+1))$ to be greater or equal than the end time of the same task (i) that takes place in unit (j) at the previous event point (n) $(T^f(i,j,n))$. In the other cases where task (i) does not take place in unit (j) at both event points (n) and (n+1), constraint (18) is relaxed. In these cases constraints (19) and (20) enforce the sequencing of the tasks.

**Sequence Constraints: Different tasks in the same unit**

$$T^s(i,j,n+1) \geq T^f(i',j,n) + t^{cl}_{jii'} wv(i,n+1) - H(1 - wv(i,n+1)) \quad \forall j \in J, i, i' \in I_j, n \in N \tag{21}$$

for tasks $(i,i')$ that are performed in the same unit (j). Note that the requirement for clean-up between tasks $(i,i')$ is explicitly considered in the above constraint thus avoiding the introduction of additional tasks to represent clean-up tasks.

**Sequence Constraints: Different tasks in different units**

If units $(j,j')$ operate in a continuous mode, then the constraints take the form:

$$T^s(i,j,n) \geq T^s(i',j',n) - H(2 - wv(i',n) - wv(i,n)) \quad \forall j, j' \in J, i \in I_j, i' \in I_{j'}, n \in N \tag{22}$$

for tasks $(i,i')$ that are performed in different units $(j,j')$ but have to be performed one after the other according to product recipe.

If unit $(j)$ operates in a continuous mode and unit $(j')$ operates in a batch mode, then the constraints take the form:

$$T^s(i,j,n+1) \geq T^f(i',j',n) - H(2 - wv(i',n) - wv(i,n+1)) \quad \forall j, j' \in J, i \in I_j, i' \in I_{j'}, n \in N \tag{23}$$

for tasks $(i,i')$ that are performed in different units $(j,j')$ but have to be performed one after the other according to product recipe.

**Sequence Constraints: Completion of previous tasks**

$$T^s(i,j,n+1) \geq \sum_{n' \in N, n' len} \sum_{i' \in I_j} (T^f(i',j,n') - T^s(i',j,n')) \quad \forall i \in I, j \in J_i, n \in N, n \neq N \tag{24}$$

representing the requirement of a task (i) to start after the completion of all the tasks performed in past event points at the same unit (j). It was found that the incorporation of these constraints enhances the computational performance of the proposed formulation.

**Objective: Maximization of profit**

$$\sum_s \sum_n \text{price}(s) d(s,n) \tag{25}$$

the objective is the maximization of production in terms of profit due to product sales. However, different objectives can also be incorporated to express different scheduling targets as for example the minimization of makespan as it was posed in the second case study presented in section 3. The minimization of the production time ($total_t$) can be posed in the following way:

$$\begin{aligned} &\min total_t \\ T^f(i,j,n) &\leq total_t \end{aligned} \tag{26}$$

## 2.1 Storage Considerations

The duration of the storage tasks are based on the rate of production and the rate of consumption of the amount of material that is stored. The following form (mathematically equivalent to the duration constraint of batch processes presented in part I[9]) is considered to represent the duration of the storage tasks ($i_{st}$) processed in unit ( $j_{st}$):

$$T^f(i,j,n) \quad = \quad T^s(i,j,n) + \alpha_{ij} wv(i,n) + \beta_{ij} B(i,j,n), \quad \forall i \in I, \quad j \in J_i, \quad n \in N$$

where $\alpha_{ij}, \beta_{ij}$ are the constant and variable term of the processing time of task (i) at unit (j) and $B(i,j,n)$ the amount of the material being stored. As mentioned above, the evaluation of $\alpha_{ij}$, and $\beta_{ij}$ terms depends on the rate of production and the rate of consumption of the amount of material that is stored. Consequently, we can distinguish three different cases concerning the operating mode of the preceding and the succeeding units.

### 2.1.1 Continuous - Continuous

#### (a) Approximation of Storage Timing

In this case, $\alpha_{ij} = 0.0$ and we can approximate $\beta_{ij}$ by the inverse of the storage rate which is considered to equal the difference between the rate of production and the rate of consumption. This approximation however, results in a lower bound on the duration of the storage task, since the consumption tasks could start after the end of the production task due to unavailability or limited capability of the consumption units. Consequently, the duration constraints for storage tasks have the following form:

$$\begin{aligned} T^f(i_{st}, j_{st}, n) - T^s(i_{st}, j_{st}, n) \quad &\geq \quad \frac{1}{rate_s(i_{st})} b(i_{st}, j_{st}, n), \\ &\quad\quad \forall i_{st} \in I_{st}, j_{st} \in J_{st}, n \in N \\ rate_s(i_{st}) \quad &= \quad rate_p - rate_c \end{aligned} \tag{27}$$

where $rate_p$, $rate_c$ are the corresponding rates of production and consumption, respectively.

For the storage tasks, in addition to the sequence constraints (22), the following constraints are incorporated to enforce that the starting times of storage tasks should coincide with the starting times of the continuous production tasks.

$$
\begin{aligned}
T^s(i_{st}, j_{st}, n) \quad &\leq \quad T^s(i', j', n) + H(2 - wv(i', n) - wv(i_{st}, n)) \\
&\forall j \in J_{st}, j' \in J, i \in I_{st}, i' \in I_{j'}, n \in N
\end{aligned} \tag{28}
$$

for storage tasks $(i_{st})$ and continuous production tasks $(i')$. If production task $(i')$ takes place in unit $(j')$ at an event point (n) then $wv(i', n) = 1$. If a storage task $(i_{st})$ also starts at the event point (n) due to the limited capability of the corresponding consumption units, $wv(i_{st}, n) = 1$ and consequently constraints (22) and the above sequence constraints become active enforcing that the starting time of storage task equals the starting time of the production task.

$$
T^s(i_{st}, j_{st}, n) \quad = \quad T^s(i', j', n)
$$

The duration constraints for storage tasks result in a lower bound of the actual duration of the task, and a second step is required for the determination of the overall schedule (as shown in Case Study 1 in section 3.1).

### (b) Improved Approximation of Storage Timings

An additional set of sequence constraints is introduced so as to make certain that the end times of the storage tasks correspond to the beginning times of the respective consumption tasks.

$$
\begin{aligned}
T^f(i_{st}, j_{st}, n) \quad &\geq \quad T^s(i_c, j_c, n+1) - H(2 - wv(i_c, n+1) - wv(i_{st}, n)) \\
&\forall j \in J_{st}, i \in I_{st}, n \in N
\end{aligned} \tag{29}
$$

$$
\begin{aligned}
T^f(i_{st}, j_{st}, n) \quad &\leq \quad T^s(i_c, j_c, n+1) + H(2 - wv(i_c, n+1) - wv(i_{st}, n)) \\
&\forall j \in J_{st}, i \in I_{st}, n \in N
\end{aligned} \tag{30}
$$

$i_c$ corresponding consumption task

$j_c$ corresponding consumption unit

expressing the requirement of the storage task $(i_{st})$ that takes place at the event point (n), (i.e., $wv(i_{st}, n) = 1$) to end by the time the consumption task $(i_c)$ begins at the next event point (n+1), (i.e., $wv(i_c, n+1) = 1$).

The duration constraints for storage tasks (27), together with the sequence constraints (28), and (29, 30) result in the exact determination of storage timings as it is shown in section 3 through the case studies applications.

Note that in the case where no direct consumption of the intermediate is allowed from production tasks, then the following set of constraints are introduced to result in the correct storage timings:

$$
\begin{aligned}
T^s(i_c, j_c, n) &\geq T^s(i_{st}, j_{st}, n) - H(2 - wv(i_c, n) - wv(i_{st}, n)) \\
&\quad \forall j \in J_{st}, i \in I_{st}, n \in N \qquad (31) \\
i_c &\quad \text{corresponding consumption task} \\
j_c &\quad \text{corresponding consumption unit}
\end{aligned}
$$

expressing the requirement of the consumption task ($i_c$) that begins at the event point (n), to start after the beginning of the corresponding storage task.

$$
\begin{aligned}
T^s(i_c, j_c, n) &\leq T^f(i_{st}, j_{st}, n) + H(2 - wv(i_c, n) - wv(i_{st}, n)) \\
&\quad \forall j \in J_{st}, i \in I_{st}, n \in N \qquad (32) \\
i_c &\quad \text{corresponding consumption task} \\
j_c &\quad \text{corresponding consumption unit}
\end{aligned}
$$

expressing the requirement of the storage task to end by the time the corresponding consumption task ($i_c$) begins at the event point (n).

### 2.1.2 Batch - Continuous

#### (a) Approximation of Storage Timing

In this case, $\alpha_{ij} = 0.0$ and the $\beta_{ij}$ can be approximated by the rate of consumption. which results in a lower bound on the duration of the storage tasks:

$$
\begin{aligned}
T^f(i_{st}, j_{st}, n) - T^s(i_{st}, j_{st}, n) &\geq \frac{1}{rate_s(i_{st})} b(i_{st}, j_{st}, n), \quad \forall i_{st} \in I_{st}, j_{st} \in J_{st}, n \in N \\
rate_s(i_{st}) &= rate_c
\end{aligned}
$$

where $rate_c$ is the corresponding rate of consumption.

The sequence constraints in this case express the requirement that the storage task starts immediately after the end of the production batch task. To enforce this requirement the following constraints are introduced in addition to constraints (23):

$$
\begin{aligned}
T^s(i_{st}, j_{st}, n+1) &\leq T^f(i', j', n) + H(2 - wv(i', n) - wv(i_{st}, n+1)) \\
&\quad \forall j \in J_{st}, j' \in J, i \in I_{st}, i' \in I_{j'}, n \in N
\end{aligned}
$$

for storage tasks ($i_{st}$) and batch production tasks ($i'$). If production task ($i'$) takes place in unit ($j'$) at an event point (n) then $wv(i', n) = 1$. If a storage task ($i_{st}$) starts at the event point (n+1) due to the limited capability of the corresponding consumption units, $wv(i_{st}, n+1) = 1$ and consequently constraints (23) and the above sequence constraints become active enforcing that the starting time of storage task equals the end time of the production task.

$$
T^s(i_{st}, j_{st}, n+1) = T^f(i', j', n)
$$

### (b) Improved Approximation of Storage Timings

The same approach suggested for the cases of continuous production and consumption tasks, in order to enforce the requirement of the end times of storage tasks to be greater or equal to the starting time of consumption tasks is also applied in this case leading to the correct storage timing.

### 2.1.3 Batch - Batch

In this case since there is no explicit rate of production and consumption of material stored, $\beta_{ij} = 0.0$ and $\alpha_{ij} = \tau_{i_{st},j_{st}}$ is considered as variable. The duration constraints for storage tasks take the following form that involves nonlinear terms:

$$\begin{aligned} T^f(i_{st}, j_{st}, n) - T^s(i_{st}, j_{st}, n) &= \tau_{i_{st},j_{st}} wv(i_{st}, n) \\ &\quad \forall i_{st} \in I_{st}, j_{st} \in J_{st}, n \in N \end{aligned}$$

where $\tau_{i_{st},j_{st}}$ is the variable duration of task $(i_{st})$ processed in unit $(j_{st})$. Employing linearization techniques[10;11], these bilinear products of continuous and binary variables can be removed. A new variable $\tau^t_{i_{st},j_{st},n}$ is defined to represent the product $\tau_{i_{st},j_{st}} wv(i_{st}, n)$ and is expressed through the following set of linear constraints:

$$\begin{aligned} \tau_{i_{st},j_{st}} - H(1 - wv(i_{st}, n)) \leq \tau^t_{i_{st},j_{st},n} \leq \tau_{i_{st},j_{st}} \quad & \forall i_{st} \in I_{st}, j_{st} \in J_{st}, n \in N \\ \tau^t_{i_{st},j_{st},n} \leq H wv(i_{st}, n) \quad & \forall i_{st} \in I_{st}, j_{st} \in J_{st}, n \in N \end{aligned}$$

Remark: It should be noted, that if the different storage requirements represent strict limitations of the intermediates to go through the storage tasks before they proceed to the next processing steps, then, these are considered in the material balances for the intermediate materials as described above. For example, as it is the case for the second case study, if the intermediates should first be stored in order to proceed to the packing line, then the material balances (11) and (12) are considered.

## 2.2 Clean-up Requirements

Clean-up requirements are considered within the sequence constraints for different tasks in the same unit (i.e., constraints (21), thus avoiding the incorporation of additional tasks. These constraints together with the sequence constraints for the same tasks at the same unit (i.e., constraints (18-20), enforce the clean-up time requirement $t^{cl}_{jii'}$ between tasks $(i), (i)'$ that are processed in unit $(j)$. In particular, consider the case that at point $(n + 1)$ task $(i)$ is processed at unit $(j)$ at which case $wv(i, n + 1) = 1$. If at the previous event point $(n)$ task $(i')$ is processed at unit $(j)$ then $wv(i', n) = 1, wv(i, n) = 0$ and consequently constraint (18) is relaxed and constraint (20) is enforced. If on the other hand at the previous event point task $(i)$ is processed, then $wv(i', n) = 0, wv(i, n) = 1$, and constraint (18) is enforced and constraint (21) is relaxed since $T^f(i', j, n) < T^f(i, j, n)$.

Note that in the cases where the clean-up tasks require the consumption of limited resources (e.g., cleaning materials, labor) the clean up tasks can be formulated as additional batch tasks with fixed duration.

In the next section, the above mathematical formulation is applied to two industrial case studies. These case studies illustrate the capability of the proposed formulation to describe and efficiently solve industrial short term scheduling problems. Storage limitations, mixed production facilities, clean-up requirements, are captured in these case studies. Furthermore, comparison with an existing approach is provided for the first case study where results are available.

# 3 Case Studies - Scheduling of Fast Moving Consumer Goods Manufacturing

The case studies presented in this section are based on an industrial fast moving consumer goods manufacturing plant. Figure 1 illustrates a sketch of the plant network. In both cases the production follows the following manufacturing route:
Making $\longrightarrow$ Storage $\longrightarrow$ Packing $\longrightarrow$ Warehousing. The first case study involves only continuous processes whereas the second case study involves batch and continuous processes. Case study 1 features 15 products, whereas case study 2 involves the production of 28 different products.

## 3.1 Case Study 1: Continuous Processes

The plant involves the production of 15 different products from 3 different bases. The making stage involves the production of 7 intermediates using 3 mixers operating in a continuous mode. The produced intermediates are then stored or packed directly in 3 storage silos and 5 packing lines operating also in continuous mode. The production rates and clean-up requirements are given in Table 1. The demands of the products at the end of the week (120 working hours) are given in Table 2.

### 3.1.1 Without Storage Constraints

First no storage limitations are considered. The STN representation of the network is shown in Figure 2 where the number in parentheses correspond to the states for the materials, and tasks for the mixing, storage and packing processing tasks. The suitable units are also shown in Figure 2 for clarity in the presentation of the network. The proposed formulation requires the consideration of 7 event points and results in a MILP model with 280 binary variables, 1089 continuous variables and 2873 constraints. Using GAMS/CPLEX for the solution of this problem requires only 60.16 CPU sec in RS-6000 workstation and the solution having and integrality gap of 0.035 was found in 180 nodes. These results outperformed the presented results for the same case study by Schilling and Pantelides[12] as illustrated in Table 3. In particular, the proposed approach requires the consideration of only 7 event

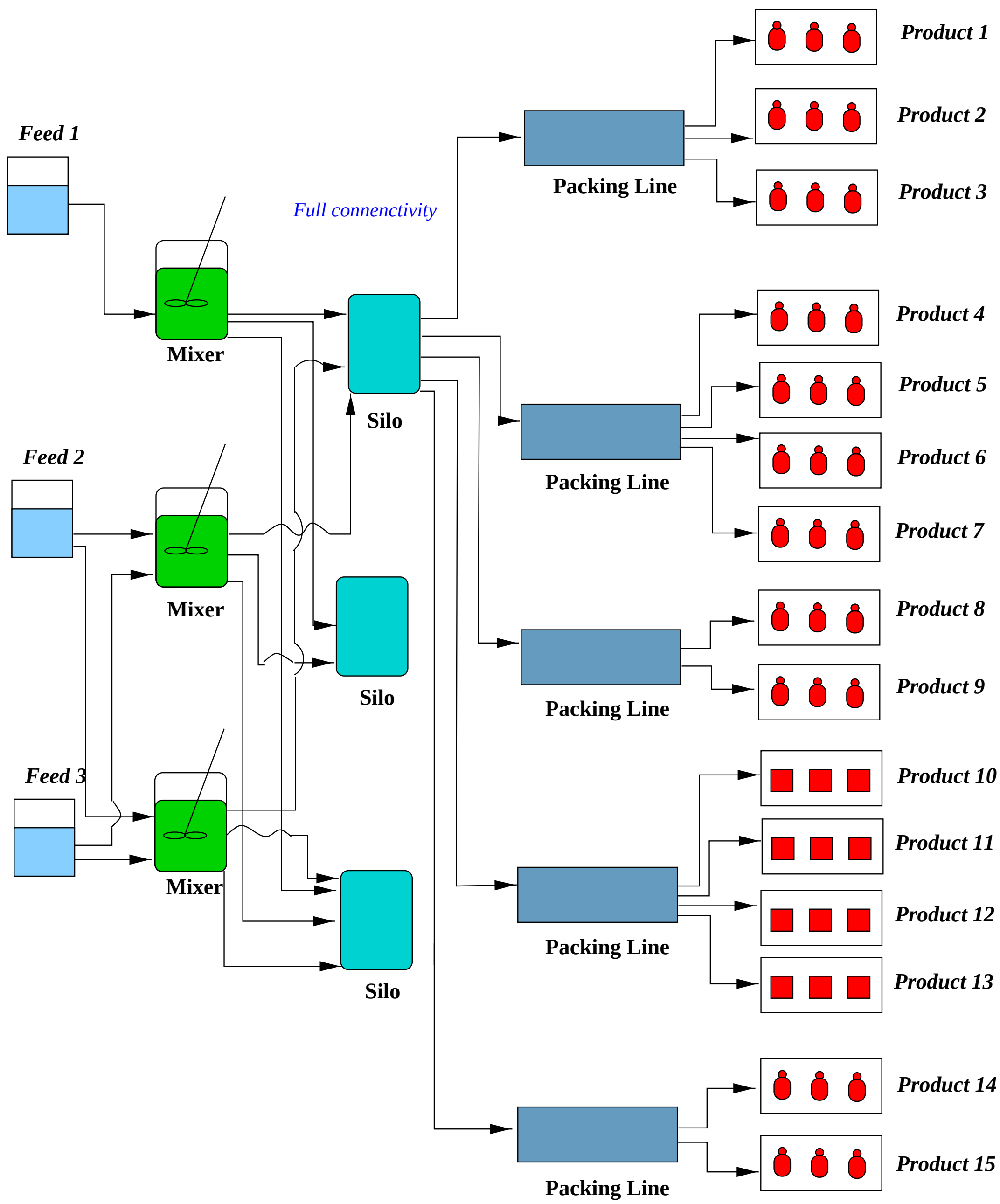


Figure 1: Plant Network

| Unit | Rate/Capacity (tn/hr or tn) | Clean-up Time (hr) |
|---|---|---|
| 1 (Mixer 1) | 17.00 | - |
| 2 (Mixer 2) | 12.24 | - |
| 3 (Mixer 3) | 12.24 | - |
| 4-6 (Silos) | 60 | - |
| 7 (Packing Line) | 5.833 | 1 |
| 8 (Packing Line) | 2.708 | 4 |
| 9 (Packing Line) | 5.571 | 1 |
| 10 (Packing Line) | 3.333(1LC) 2.241(0.5LC) | 2 |
| 11 (Packing Line) | 5.357 | - |

Table 1: Unit capacities and Clean-up requirements

| Product | Demand | Product | Demand |
|---|---|---|---|
| P1 | 220 | P9 | 13.5 |
| P2 | 251 | P10 | 114.5 |
| P3 | 15 | P11 | 53 |
| P4 | 116 | P12 | 16.5 |
| P5 | 7 | P13 | 8.5 |
| P6 | 47 | P14 | 2.5 |
| P7 | 144 | P15 | 17.5 |
| P8 | 42.5 | | |

Table 2: Required Product Demands

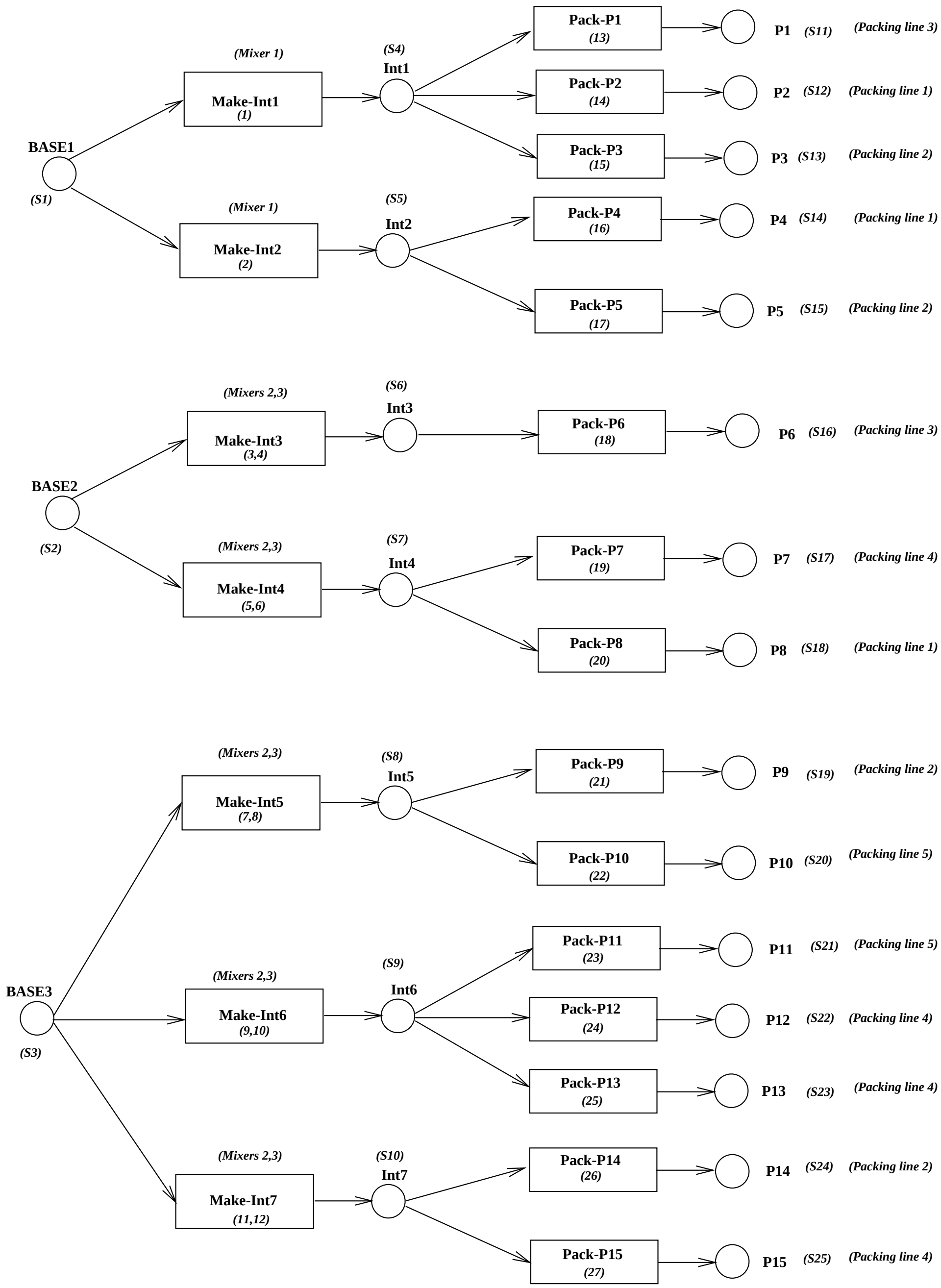


Figure 2: State Task Network Representation of the Plant

points compared to 30 points used by Schilling and Pantelides[12]. Moreover, note that the formulation of Schilling and Pantelides[12] does not guarantee that 30 points are sufficient to obtain the optimal schedule since the consideration of 35 points leads to better objective function of 2612 units but still worse than the profit found from the first integer solution of the proposed formulation (2626.57) that corresponds to the optimal schedule within 0.035 integrality tolerance. The final optimal integer solution found with the proposed approach is 2689.42 as shown in Table 3. The proposed formulation results in a MILP problem having 280 binary variables, 1089 continuous variables and 2873 constraints compared to 1042 binary variables, 2746 continuous variables and 4981 constraints needed in the Schilling and Pantelides[12] formulation. Due to its smaller size and smaller integrality gap 0.013 compared to 0.044, the proposed model requires significantly less CPU time to be solved, 60.16 CPU seconds per iteration in a single workstation RS-6000, compared to 3407 CPU sec needed by the parallel implementation of the approach of Schilling and Pantelides[12] in which 6 processors were utilized. The optimal schedule is shown in Figure 3.

| | Schilling and Pantelides[12] | Proposed Formulation | |
|---|---|---|---|
| | | First Integer | Optimal Solution |
| No of event points | 30 | 7 | 7 |
| Integer Variables | 1042 | 280 | 280 |
| Continuous Variables | 2746 | 1089 | 1089 |
| Constraints | 4981 | 2873 | 2873 |
| MILP Solution | 2604 | 2626.57 | 2689.42 |
| LP Relaxation | 2724 | 2724 | 2724 |
| Integrality Gap | 0.044 | 0.035 | 0.013 |
| Nodes | 91 | 381 | 9 Iterations |
| CPU time (sec) | 3407* | 60.16** | $\approx 9 \times 60 = 540$ |

$*$ parallel implementation using 6 processors
$**$ single workstation RS-6000

Table 3: Comparison of Results: Case Study 1

### 3.1.2 Incorporation of Storage Constraints

Then storage limitations are also considered based on the ideas of section 2.1. The STN representation of the production network in this case is shown in Figure 4 where the number in parentheses correspond to the states for the materials, tasks for the mixing, storage and packing processing tasks, and the suitable units.

**(a) Approximation of storage timings**

As shown in Table 4, the proposed formulation in this case requires also the consideration of 7 event points and results in a MILP model having 336 integer variables, 1361 continuous variables and 3395 constraints. The solution of the resulting MILP problem using

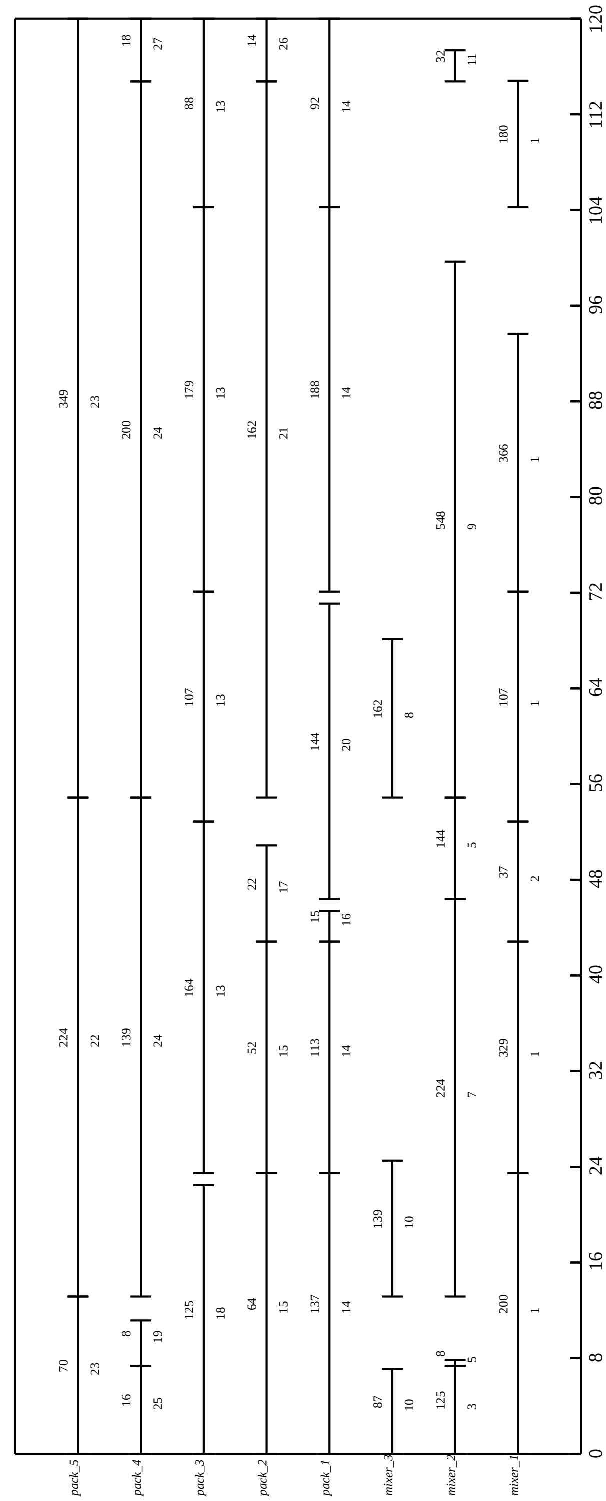

Figure 3: Optimal Schedule for Case Study 1 : No Storage Consideration

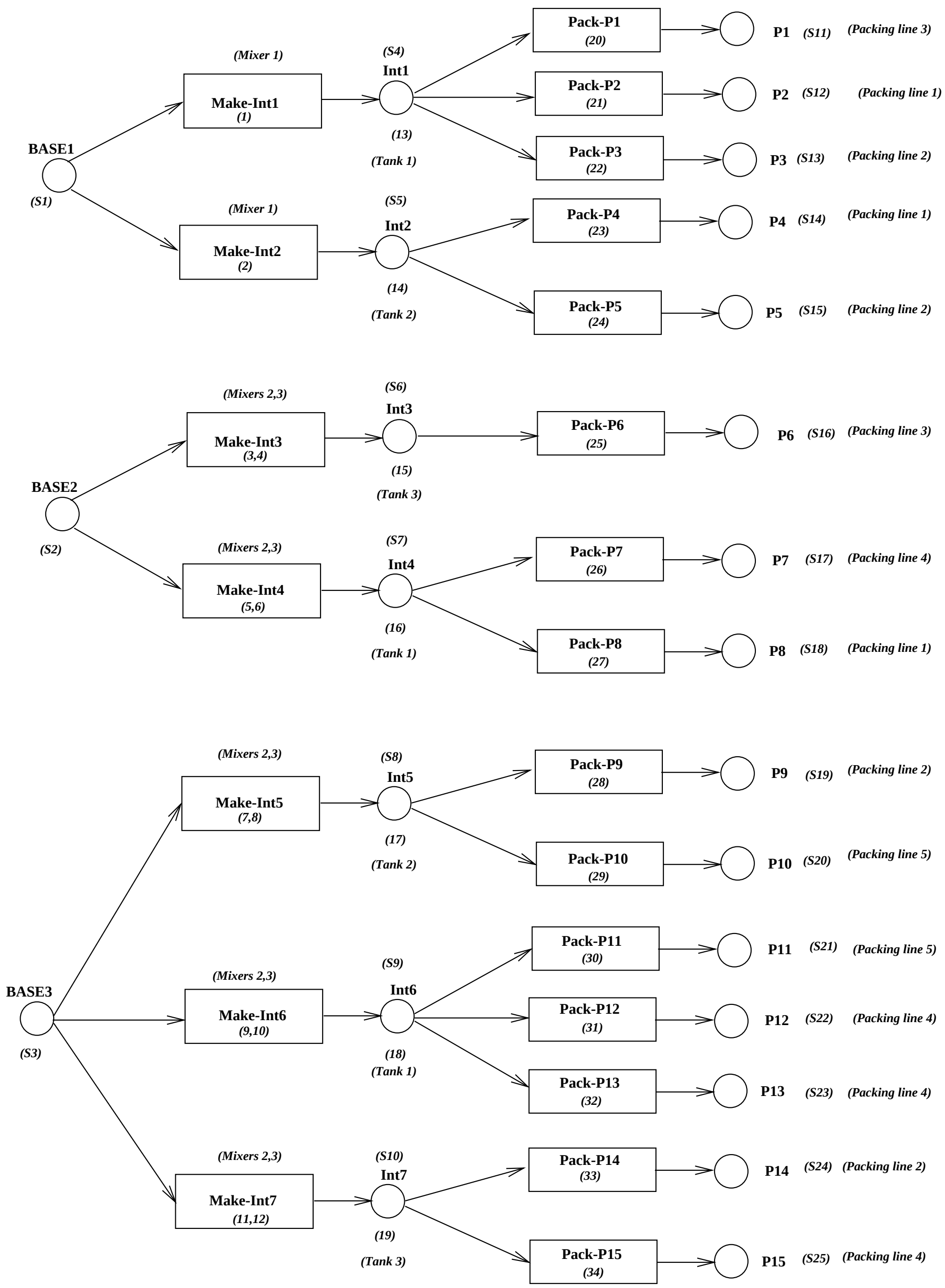


Figure 4: State Task Network Representation of the Plant

| | Proposed Formulation | |
|---|---|---|
| | First Integer | Optimal Solution |
| No of event points | 7 | 7 |
| Integer Variables | 336 | 336 |
| Continuous Variables | 1361 | 1361 |
| Constraints | 3395 | 3395 |
| MILP Solution | 2649.56 | 2688.24 |
| LP Relaxation | 2724 | 2724 |
| Integrality Gap | 0.027 | 0.013 |
| Nodes | 197 | 6 Iterations |
| CPU time (sec) | 66.8 | $\approx 6 \times 70 = 420$ |

Table 4: Storage Consideration: Case Study 1

GAMS/CPLEX in RS-6000 workstation requires 66.8 CPU sec and the exploitation of 197 nodes and corresponds to the optimal profit of 2649.56 units. Following the iterative procedure described above a better solution is obtained that corresponds to an optimal schedule with profit 2688.24 which is within 0.013 from the best integer solution. The optimal schedule is shown in Figure 5 and Figure 6 after the modification of the storage timing. In particular, the storage of 6 units of intermediate 1 (Int1) beginning at time 0 should be prolonged until the time 48.681 hr when is consumed from packing line 3 to produce product P1. 60 units of Int1 stored at 49.681 hr should remain at the storage tank 1 until 77.169 hr when 54 units are processed in packing line 2 to produce product P3 and 96.266 hr when the rest 6 units are processed in packing line 3 to produce product P1. Int2 should also stay at the second storage tank from 46.151 hr until 55.015 hr when is consumed in packing line 2 to produce product P4. Int 6 should be stored from 97.266 hr to 111.564 hr when is consumed by packing line 4 to produce product P13 and finally, 4 units of intermediate 7 should be stored from 102.518 hr to 104.518 hr when is consumed in packing line 4 to produce product P15. It should be noted that in all cases the proposed formulation determines the assignment of storage units to storage tasks, the exact starting times of storage tasks and the amounts of the intermediates that need to be stored.

**(b) Improved Approximation of storage timings**

Following the approach described in section 2.1.1, the resulting mathematical model involves 3260 constraints, 1337 continuous and 360 binary variables and requires 8 iterations to obtain the optimal schedule of profit equal to 2695.25 within 0.01% integrality tolerance in a RS-6000 workstation. The optimal schedule is shown in Figure 7.

## 3.2 Case Study 2: Batch and Continuous Processes

The data for this case study are given in Tables 5, 6. The plant involves the production of 28 different products using 3 bases through a sequence of mixing storage and packing stage.

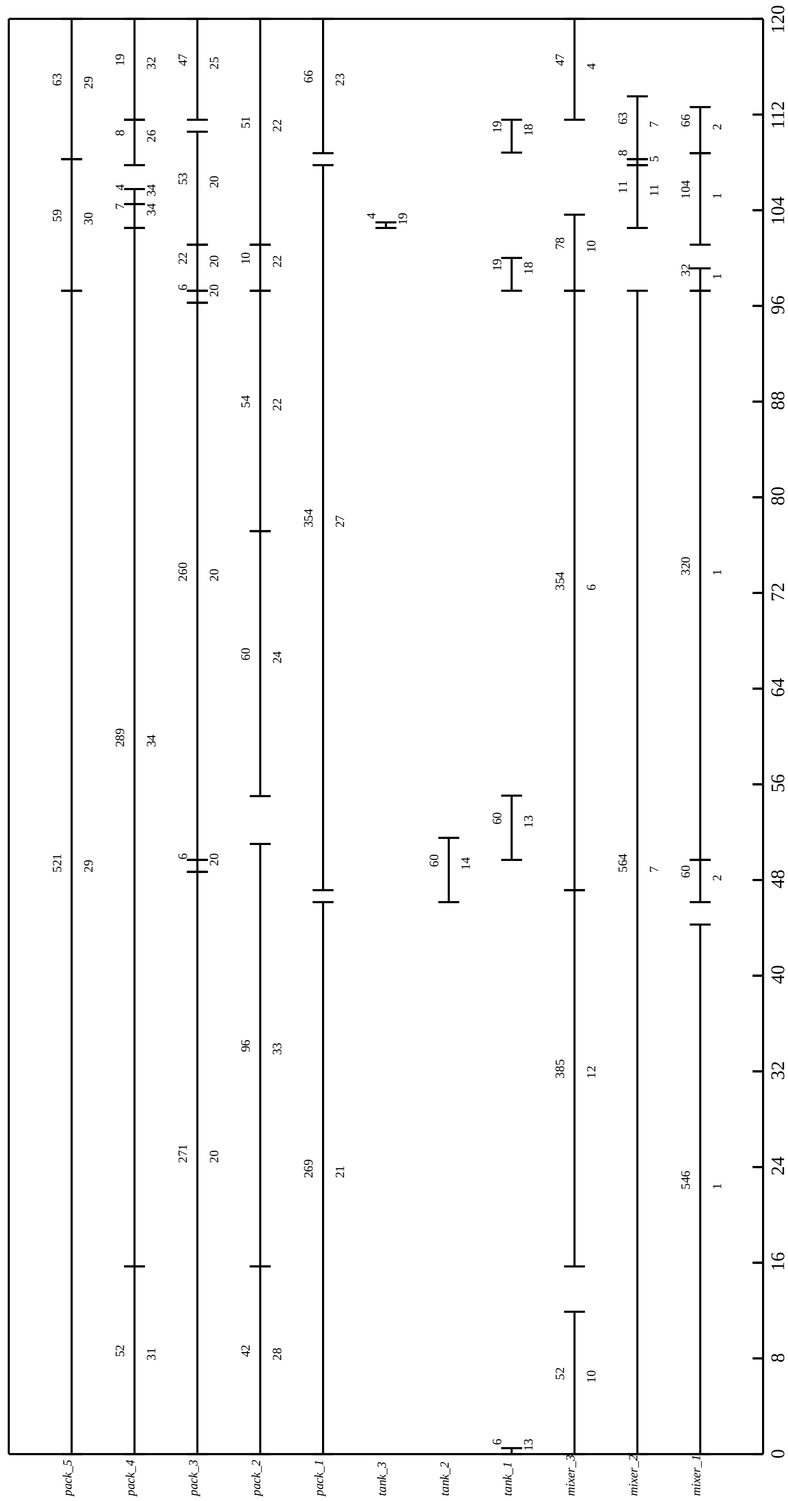

Figure 5: Optimal Schedule for Case Study 1 : Approximate Storage

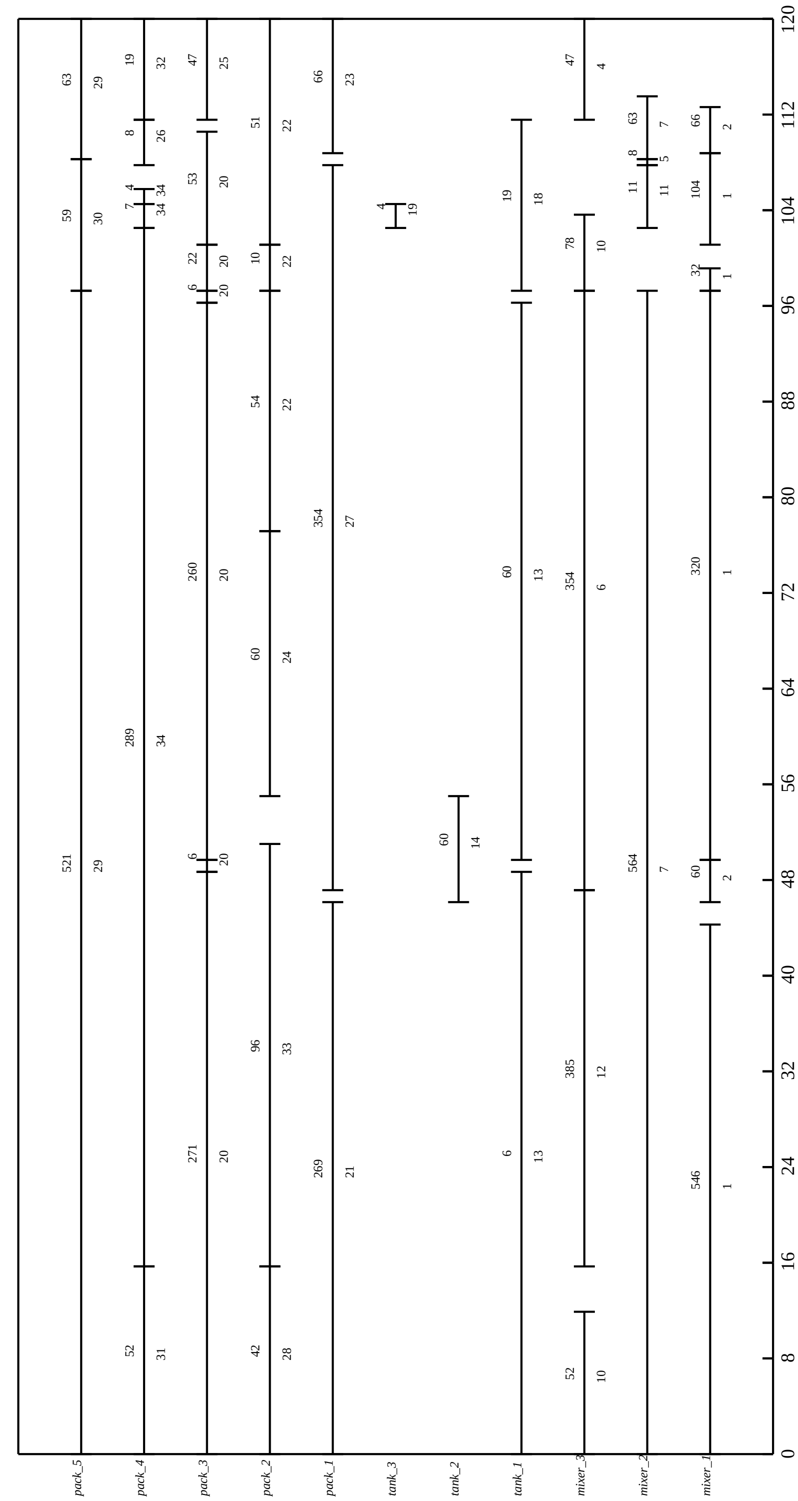

Figure 6: Optimal Schedule for Case Study 1 : Corrected Approximate Storage

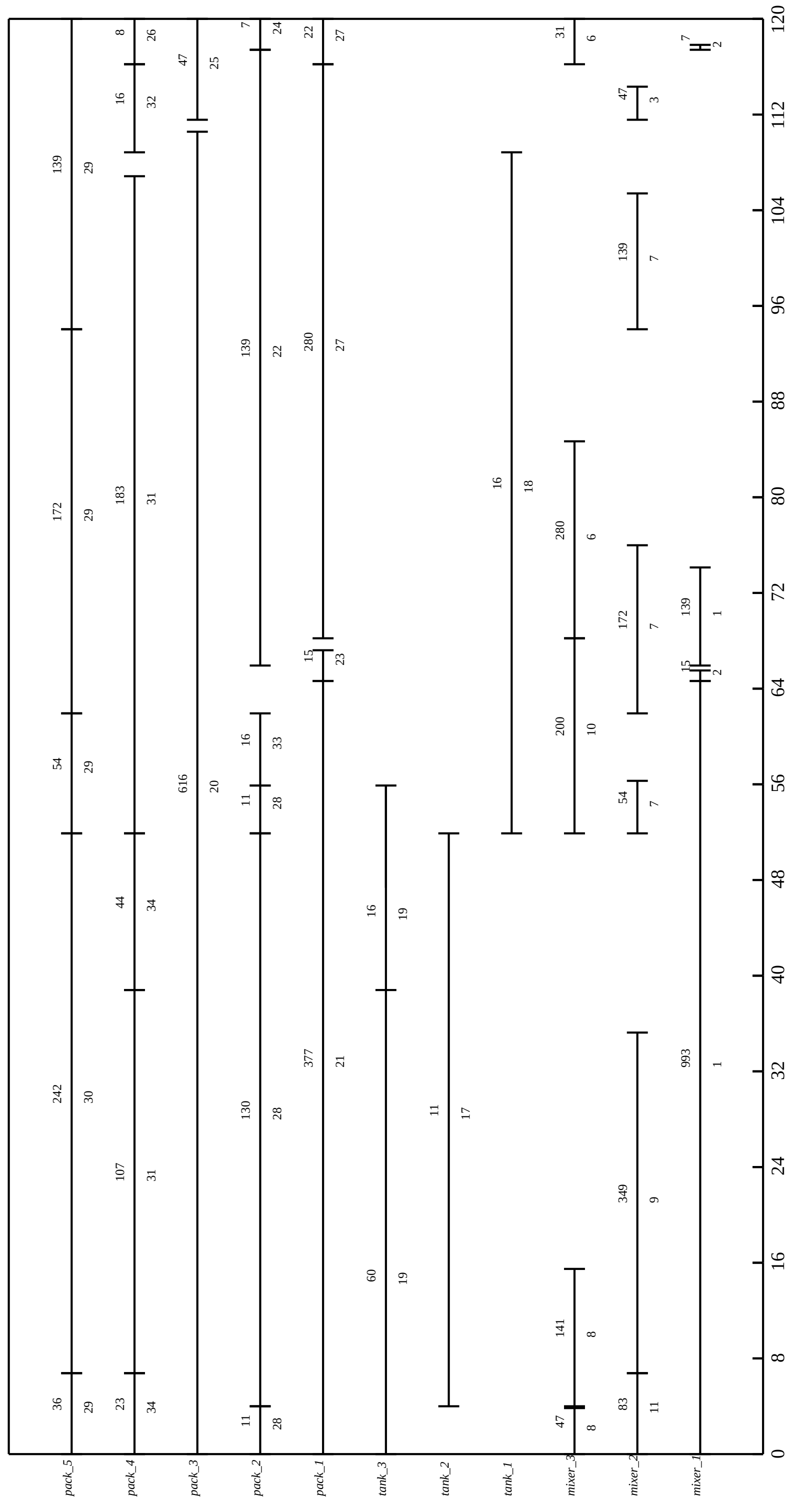

Figure 7: Optimal Schedule for Case Study 1 : Improved Storage Timings

The first base corresponds to the dilute base denoted as $D_Q T$ and is processed in the first mixer that operates in a continuous mode to produce 3 different intermediates that are then stored and packed in packing lines 1, 2, and 3 to produce 8 final products. The second dilute base and the third concentrated base denoted as $D_H Q$ and $C_H Q$, respectively, are processed in the second batch mixer to produce 7 different intermediates to be stored and packed in packing lines 1, 2, 3, 4 and 5 to produce 20 final products, 3 from the dilute base and 17 from the concentrated base. The intermediates should first be stored at one of the 5 storage silos that are available having a maximum capacity of 60 tn, before they proceed to the one of the 5 packing lines. As soon as a packing line becomes available the intermediates proceed to the packing stage where they are packed in different packing sizes and types to generate the required products. Packing line 1 produces 2(L) liter of dilute product, packing line 2 produces 1L bottles of both dilute and concentrated products, packing line 3 produces 4L bottles of dilute and concentrated products, packing line 4 produces cartons of 1L or 0.5L of dilute and concentrated products and packing line 5 packs 1L of concentrated products. The plant works 120 hr a week in 3 shifts of 8 hr per day.

| Product | Demand | Product | Demand |
|---|---|---|---|
| P1 | 220 | P15 | 45.5 |
| P2 | 251 | P16 | 53 |
| P3 | 116 | P17 | 2.5 |
| P4 | 17 | P18 | 16.5 |
| P5 | 93 | P19 | 10 |
| P6 | 6 | P20 | 3 |
| P7 | 15 | P21 | 14.5 |
| P8 | 7 | P22 | 5 |
| P9 | 47 | P23 | 13.5 |
| P10 | 8.5 | P24 | 17.5 |
| P11 | 144 | P25 | 7 |
| P12 | 42.5 | P26 | 33 |
| P13 | 114,5 | P27 | 3 |
| P14 | 3 | P28 | 11 |

Table 5: Required Product Demands

The rates and capacities of the units are given in Table 6 where 1LC means the production of 1lt cartons, 0.5LC denotes 0.5lt cartons, and D, C stands for dilute and concentrate, respectively. For the storage of the intermediates there are 5 silos available each one having capacity of 60tn. In both making and packing production stages there are clean-up requirements the duration of which is sequence dependent. The required clean-up times are given in Table 6.

Given this information, the objective is to determine the schedule that minimizes the overall production time as in the form (26) and meets the required demand of the products. In the sequel, the problem without storage requirements is considered first, and then storage constraints are also incorporated.

| Unit | Rate/Capacity (tn/hr or tn) | Clean-up Time (hr) |
|---|---|---|
| 1 (Continuous Mixer) | 20 | 0.5 |
| 2 (Batch Mixer) | 15 | 2.25 (D→C) |
| | | 2.25 (D↔D) |
| | | 2.25 (C↔C) |
| 3-8 (Silo) | 60 | - |
| 9 (Packing Line) | 5.833 | - |
| 10 (Packing Line) | 2.708 | 36 |
| 11 (Packing Line) | 5.571 | - |
| 12 (Packing Line) | 3.333(1LC) 2.241(0.5LC) | 2 (1LC↔0.5LC) |
| 13 (Packing Line) | 5.357 | - |

Table 6: Unit capacities and Clean-up requirements

### 3.2.1 Without Storage Constraints

Based on the given data the STN representation of the plant without considering storage tasks and silos is shown in Figure 8.

The problem involves 41 states (raw materials, intermediates, products), 7 units (mixers, packing lines), and 38 different processing tasks(mixing, packing). The resulting MILP model has 1900 binary variables representing the assignment of the processing tasks to event points, as well as the assignment of units to event points and 9153 continuous variables representing the beginning and end times of processing tasks, the amount of material being processed and the amount of products being produced. Note that due to the small capacity and processing time of the batch mixer, at least 45 event points are needed in order to satisfy the product demands since 7 intermediates are produced by this mixer.

The solution of the resulting model required 131 linear programming problems to be solved in 3666 CPU sec in a HP-C160 workstation to get the first integer solution of 111.317 hr. The optimal solution is obtained at the second iteration and corresponds to the production time of 111.108 hr. The resulting schedule is shown in Figure 9. Note that the overall production time was reduced to 111.108 hr instead of 120 hr. It is found that the incorporation of a penalty term in the objective function penalizing the utilization of the units enhances the computational efficiency of the algorithm and does not affect the quality of the solution since the demand satisfaction is posed as equality constraints and consequently are exactly satisfied in any feasible solution.

### 3.2.2 Incorporation of Storage Constraints

The incorporation of storage requirements involves the consideration of 5 storage silos, where the intermediates are stored first before proceeding to packing lines. This requires the consideration of 7 additional tasks for the storage of each intermediate. Packing line 2 has the requirement of 36 hr clean-up time between the processing of dilute and concentrate products which allows for only one change over between dilute and concentrate within the time

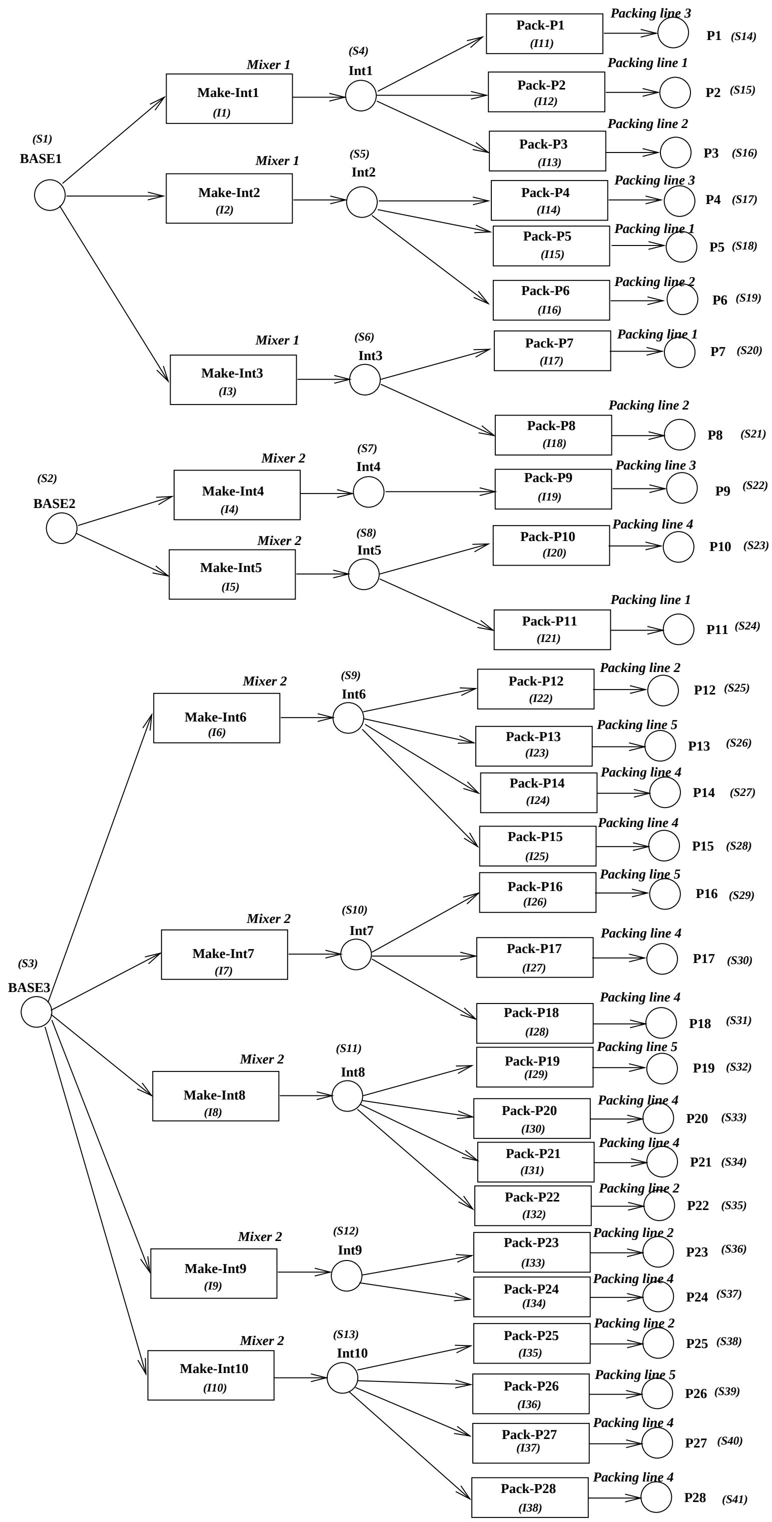


Figure 8: State Task Network Representation of the Plant

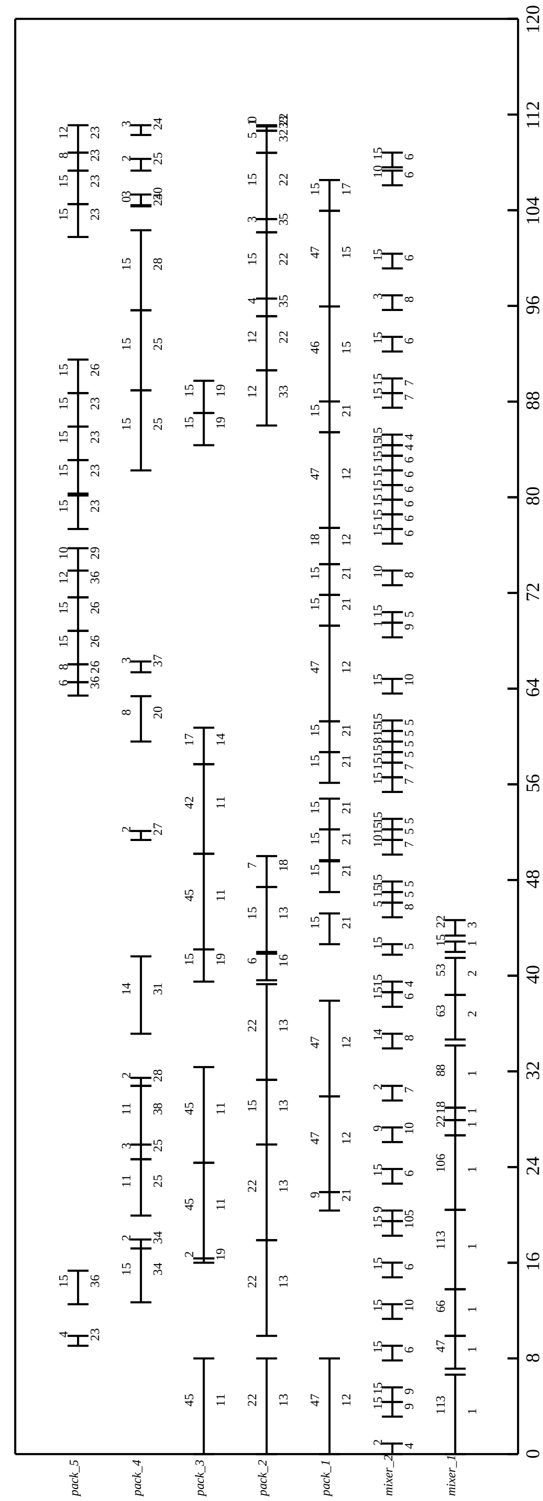


Figure 9: Optimal Schedule without Storage Consideration: Case Study 2

horizon. This fact that comes from the physical description of the system allows the problem decomposition into two parts: (a) the first part that involves the first continuous mixer that produces 3 intermediates which are stored in three silos and then proceed to the three corresponding packing lines to produce products P1 to P8; and (b) the second part which involves the batch mixer 2, 5 silos and 5 packing lines. Notice that by considering this decomposition there is no longer the requirement of 36 hr clean up time since the first continuous mixer produces the dilute intermediates and the second batch mixer produces the concentrate ones.

*First Part:*

As mentioned above the first part involves 1 continuous mixer, 3 silos and 3 packing lines. The STN representation of this part is shown in Figure 10 where the assignment of the units to tasks is also indicated.

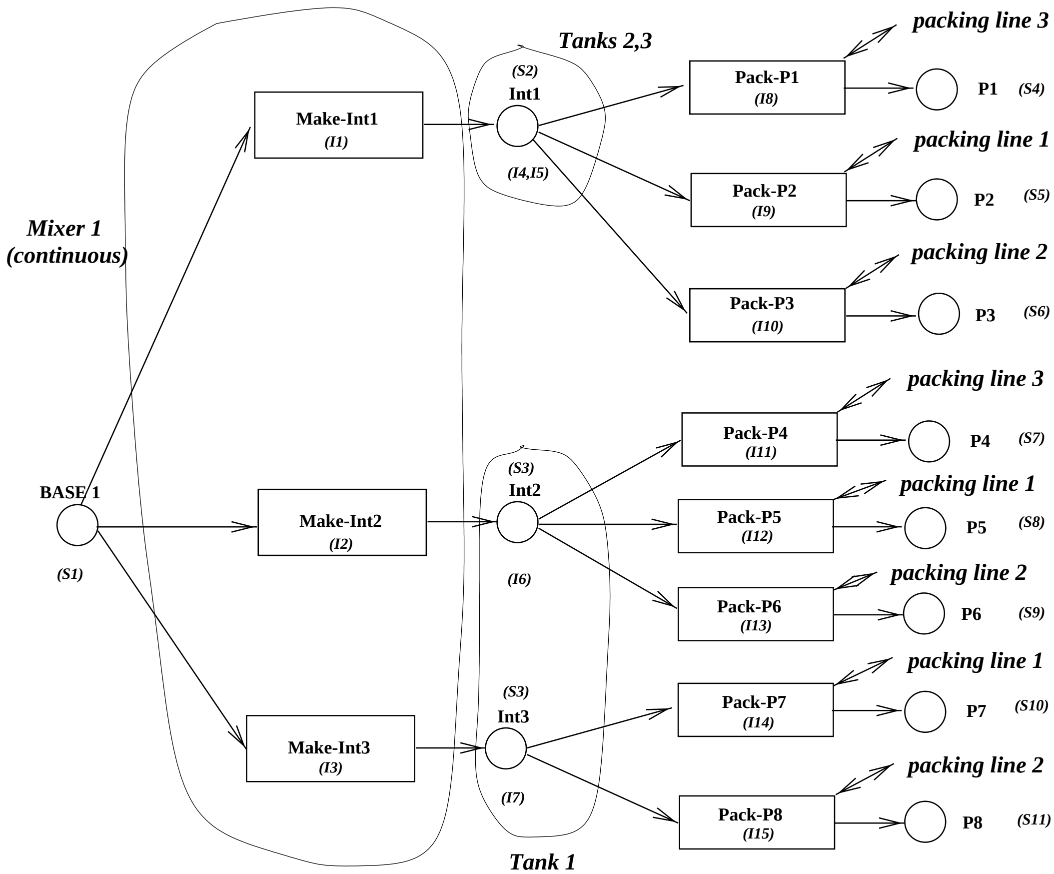


Figure 10: STN Representation of the First Part of the Plant: Case Study 2

A penalty term of the form $\sum_{i,n}(penalty_i)wv(i,n)$ is incorporated in the objective function to penalize the utilization of the units in order to improve the computational performance of the solution approach. Note that a penalty term consideration does not affect the schedule production, since the product demands satisfaction are posed as hard constraints in this case study. The resulting model involves 264 binary variables, 915 continuous variables and 1904 constraints and is solved using GAMS/CPLEX in a HP-C132 workstation. It requires the solution of 54 LP relaxations in 17.5 CPU sec for the first integer solution of 62.728 hr production time to be obtained that corresponds to an integrality tolerance of 0.99%. However, employing the iterative procedure described previously the final solution of 61.546 hr is achieved after 3 iterations. The optimal solution is within 0.46% of the best integer solution and corresponds to the schedule illustrated in Figure 11.

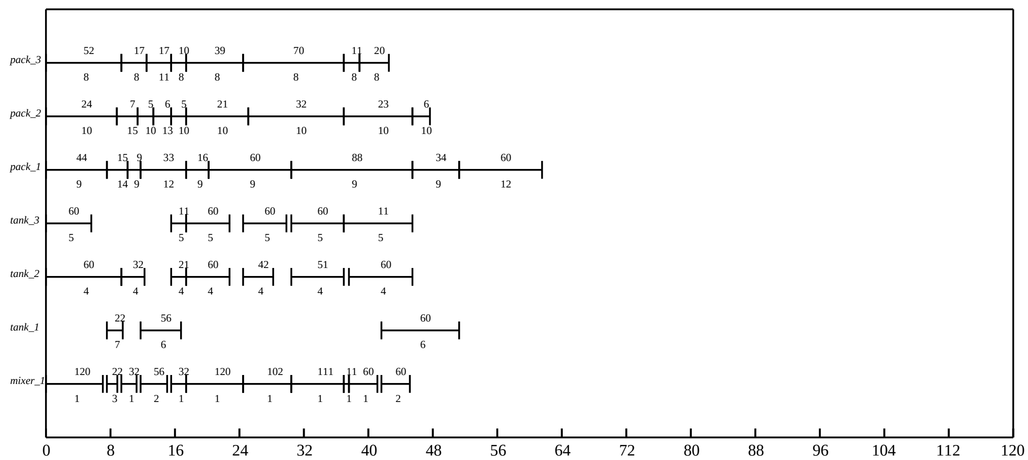


Figure 11: Optimal Schedule for the First Part of the Plant: Case Study 2

*Second Part:*

The second section of the plant involves the batch mixer, 5 silos and 5 packing lines and produce 20 products. The State Task network representation of this part is shown in Figure 12.

The mathematical formulation for this problem requires the consideration of 45 event points due to the small capacity of batch mixer and involves 2375 binary variables and 29384 continuous variables. The first integer solution of the MILP model using GAMS/CPLEX requires the solution of 590 LP relaxation problems and is solved in 9890 CPU sec in a HP-C160 workstation and correspond to the production time of 111.676 hr. The optimal schedule for this part, shown in Figure 13, is found in the second iteration and correspond to the production completion in 111.52 hr instead of 120 hr, that represents a more than a whole shift reduction. Note, that the solution of the second part of the plant requires the additional timing constraints for the packing lines based on the solution of the first part so that the overall schedule is feasible. That includes the following constraints:

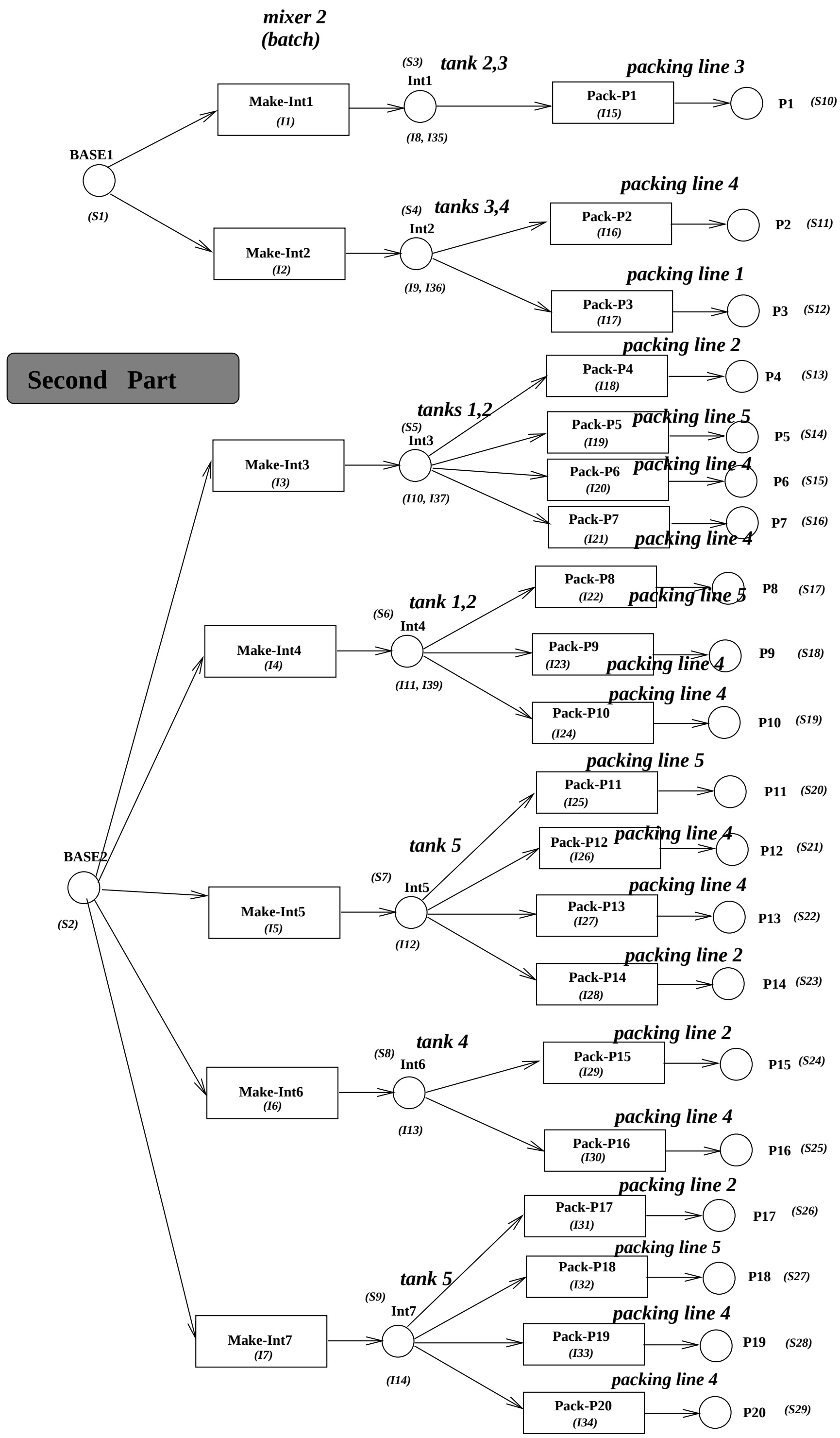


Figure 12: STN Representation of the Second Part of the Plant: Case Study 2

$$T^s(i, j_7, n) \geq 61.546 \tag{33}$$

$$T^s(i, j_8, n) \geq 83.637 \tag{34}$$

$$T^s(i, j_9, n) \geq 42.542 \tag{35}$$

where constraint (23), (24) and (25) corresponds to packing line 1, 2 and 3 respectively, that are already partially occupied from the solution of part 1.

Remark: It should be noted that the suggested decomposition relies on the nature of the case study and the specified clean-up time for packing line 2. Hence, only two alternatives have to be considered for the solution of the overall problem, the first one examined before where the dilute products are processed first and the second one where the concentrate products are packed first which gives an infeasible scheduling problem. Consequently, the schedule obtained above following the plant decomposition corresponds to the optimal solution for the overall plant.

# 4 Conclusions

In this paper, the novel continuous-time formulation for short term scheduling presented in part I was extended to consider continuous processes, as well as mixed production facilities involving batch and continuous processes. It is shown that the proposed formulation is capable of handling efficiently limited storage requirements by considering storage tasks as additional batch tasks with variable processing times. Moreover, clean-up requirements are incorporated by considering clean-up times in the timing sequence constraints, thus avoiding the consideration of additional tasks. It was shown that the proposed formulation can address large-scale industrial problems. Two case-studies were presented involving the production of 15 and 28 products. Comparisons with the existing approaches illustrates that the presented formulation results in simpler models where real plant characteristics can be easily accommodated. Moreover, the proposed approach offers the flexibility of exploiting the problem's special structure and thus being able to solve it very efficiently.

**Acknowledgments**
The authors gratefully acknowledge financial support from the National Science Foundation and Mobil Technology Company. The authors also thank Dr. Tahmassebi of Unilever for providing the second industrial case study.

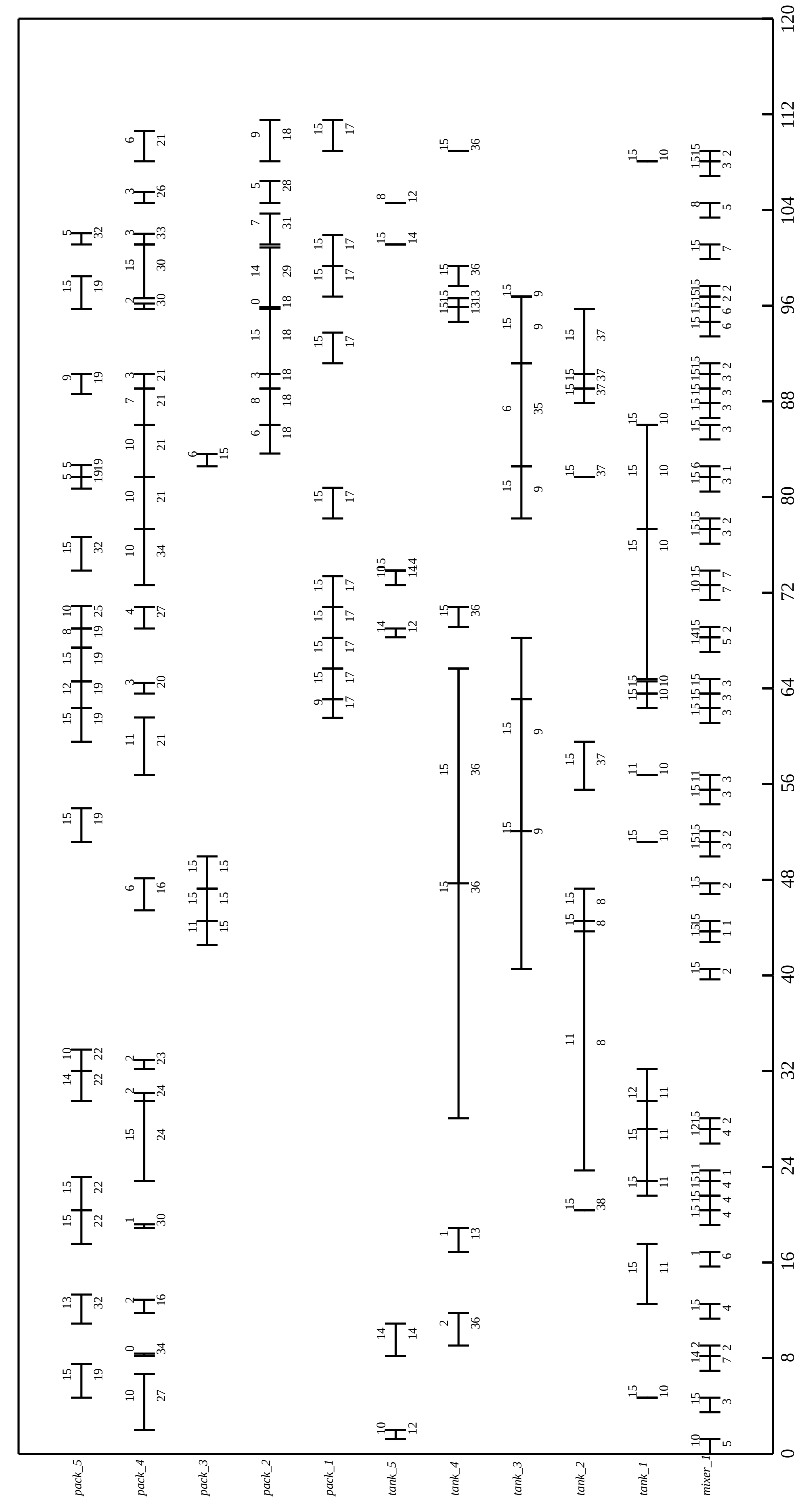


Figure 13: Optimal Schedule for the Second Part of the Plant: Case Study 2

# References


(1) Sahinidis, N. V.; Grossmann, I. MINPL Model for Cyclic Multiproduct scheduling on Continuous Parallel Lines. *Comp. Chem. Engng.* **1991**, *15*, 85.

(2) Pinto, J.; Grossmann, I. Optimal Cyclic Scheduling of Multistage Continuous Multiproduct Plants. *Comp. Chem. Engng.* **1994**, *18*, 797.

(3) Kondili, E.; Shah, N.; Pantelides, C. C. Production Planning for the Rational Use of Energy in Multiproduct Continuous Plants. *Comp. Chem. Engng.* **1993**, *17*, S123.

(4) Zhang, X.; Sargent, R. The Optimal Operation of Mixed Production Facilities - Extensions and Improvements. *Comp. Chem. Engng.* **1996**, *20*, S1287.

(5) Schilling, G.; Pantelides, C. C. A Simple Continuous-time Process Scheduling Formulation and a Novel Solution Algorithm. *Comp. Chem. Engng.* **1996**, *272*, 1001.

(6) Karimi, I.; McDonald, C. Planning and Scheduling of Parallel Semi-Continuous Processes - Part II : Short-Term Scheduling. *Ind. Eng. Chem. Res.* **1997**, *36*, 2701.

(7) McDonald, C.; Karimi, I. Planning and Scheduling of Parallel Semi-Continuous Processes - Part I : Production Planning. *Ind. Eng. Chem. Res.* **1997**, *36*, 2691.

(8) Kondili, E.; Pantelides, C. C.; Sargent, R. A General Algorithm for Short-Term Scheduling of Batch Operations - I. MILP Formulation. *Comp. Chem. Engng.* **1993**, *17*, 211.

(9) Ierapetritou, M. G.; Floudas, C. Effective Continuous-Time Formulation for Short-Term Scheduling: I. Multipurpose Batch Processes. *Ind. Eng. Chem. Res.* **1998**, this issue.

(10) Floudas, C. *Nonlinear and Mixed-Integer Optimization*; Oxford University Press, 1995.

(11) Glover, F. Improved Linear Integer Programming Formulations of Nonliear Integer Problems. *Management Science* **1975**, *22*, 455.

(12) Schilling, G.; Pantelides, C. C. A Hybrid Branch and Bound Algorithm for Continuous-Time Process Scheduling Formulations. *AIChE Annual Meeting* **1996**, , paper No 171d.