From Scheduling to Supervision in Batch Processes

R. Champagnat¹,² P. Esteban¹,² H. Pingaud³ R. Valette¹

1: LAAS-CNRS UPR 8001, 7 avenue du colonel Roche, F-31077 Toulouse Cédex 4, France
2: Université Paul Sabatier, 118 Route de Narbonne, F-31062 Toulouse Cédex 4, France
3: Laboratoire de Génie Chimique UMR CNRS 5503, ENSIGC, chemin de la loge, F-31078 Toulouse Cx
Tel: +33 (0) 561 336 416
Fax: +33 (0) 561 336 936
E-mail: champa@laas.fr

ABSTRACT

This paper deals with the supervision of batch processes. Supervision is in charge of the plan execution in real-time, according to the actual state of the plant. We consider two models commonly used for modeling the supervision level and the scheduling level: Petri nets and activities-on-arcs graphs. After a brief presentation of these models and the way they are used for supervision and scheduling, this paper compares their ability to detect inconsistent sequences. Three kinds of inconsistencies are considered: structural, decisional and quantitative. For each one an illustrative example is given.

1. INTRODUCTION

Chemical industries have recently changed their production mode. The globalization of market and the quick changes of the needs of the society generate a new industrial strategy. This strategy, based on multi-product/multipurpose batch plants, allows to concentrate the production on one site; it also allows to produce a wide range of products on a plant, without any modification [3, 8, 15, 18]. But this production mode, which is really attractive, is also very complex to implement.

A batch control system can be refined into five levels: planning, scheduling, supervision, coordination and local control. The planning level determines the production strategy according to an aggregate representation of the plant. The scheduling level must determine the sequence of operations in the plant according to the resource capacities. The aim of supervision is to implement real-time scheduling according to the resource availabilities and the actual state of the system [5, 10]. The coordination level realizes the coordination of subsystems such as transportation. And the local control implements the real-time control. As there are different products, there are different recipes. Thus scheduling level becomes more complex. And the role of the real-time monitoring increases. Moreover, the operation duration may vary and the plan has frequently to be modified.

The supervision requires to know the short term scheduling and the actual current state of the equipments each time a decision is made. Short term scheduling can be modeled by an “activities on arcs” (AOA) graph, and supervision by Petri nets (PN). This paper presents briefly these two models.

Then this paper discusses about the supervision. The main goal is to avoid critical situations (deadlock). But as in batch systems the transfer and the buffer capacity cannot be neglected, logical constraints involved are very important. A first example shows how the two models (AOA-graph and PN) can help the supervision level to deal with these constraints. As backtracking is not possible in real-time control, a second example deals with inconsistent decisions. And a third example deals with the way of meeting time constraints.

2. MODELS FOR SCHEDULING AND SUPERVISION

A Petri nets

Petri nets are commonly used to model the supervision level in manufacturing systems. Their advantages are that they allow formal analysis, efficient discrete event simulation and can directly be implemented [17].

Typically, places represent operations or resource states and transitions describe events, that are state changes. It is also possible to attach operations to transitions. It is an abbreviation. A transition corresponds to a sequence of a transition (beginning of the operation), a place (operation) and a transition (end of the operation). The time is attached to the aggregated transition.

In batch systems, recipes generally represent a sequence of operations. Typically, Petri net model of a batch system is obtained by a composition of sub-nets representing recipes of various products, with other subnets representing the resource allocation mechanisms. Frequently it is possible to obtain a Petri net of the class called $S^2$PR or a net that can be reduced to a net of this class [11, 6, 2].

In the case of the aggregated representation, the durations of the operations are attached to transitions. The Petri nets used for supervision are typically repetitive. They do not describe a plan execution along some time horizon. They have to be controlled, this means that the firing of the transitions corresponding to the beginning of the operations will be triggered by the plan. Another important point to be underlined is that a place or a transition does not represent a unique occurrence of an operation, they represent a type of operation which will be cyclically executed.

Petri nets can also be used at the scheduling level to elaborate the production plan. Typically it is in the case of cyclic policies and when operations are executed as soon...
B Activity-on-arc graphs

Scheduling production systems when the policy is not cyclic, may be based on "activity-on-arc graphs" (AOA-graphs). An AOA-graph is composed of nodes and oriented arcs [16]. The operation durations are associated with arcs. The beginning and the end of the operations are attached to nodes. The precedence relations between the operations of a recipe are represented in a quantitative way by arcs [4, 7, 9, 14].

An operation duration is represented by an arc connecting its beginning node to its end node. The value of the duration is attached to the arc as a label. It has to be pointed out that a node corresponds to a unique operation instance. It will be "active" once and only once during the whole time horizon over which the plan has been elaborated. A precedence relation between two operations \( O_1 \) and \( O_2 \) is denoted by an arc connecting the end node of \( O_1 \) to the beginning node of \( O_2 \). The duration of the time interval between the two corresponding events is attached to the arc as a label. When it is just a precedence relation between two operations (say \( O_1 \) and \( O_{i+1} \)) on the same batch and without delay, it is possible to merge the end node of \( O_1 \) with the beginning node of \( O_{i+1} \) in order to have a more compact representation. It has to be pointed out that this aggregation differs from that which is usual in Petri nets where it is the beginning and the end transition which are merged.

Typically AOA-graphs are used to compute the optimal schedule (optimal plan) on a time horizon. It can also be used to characterize the set of schedules which respects all the constraints (operation durations, precedence constraints and due dates).

AOA-graphs can be used for supervision in the following way. Two graphs are required. The first one represents the remaining operations and the resource constraints and the second one, the plan. When an operation is finished the two graphs are updated. If the current state is not the projected one and if the plan no longer respect the constraints, a new plan is generated.

C Inconsistency of a set of precedence constraints

Supervision is in charge of implementing the provisional plan in real-time. It would seem simpler to use the same model at the two levels (scheduling and supervision), however, typically AOA-graphs are used for scheduling and Petri nets for supervision. The issue is then: are Petri net really more adequate for supervision and AOA-graph for scheduling? Do these advantages compensate the fact that the interface between the two levels will be more difficult to implement because of the difference of the models?

In order to understand better the benefits and the disadvantages of these two approaches, we will now analyze how they allow the detection and the avoidance of constraints violations. Three kind of inconsistencies can be differentiated: structural, decisional and quantitative.

If in a Petri net the current marking is dead because some deadlock has been reached, this means that the sequence of transition firings can no longer be extended. This means that any plan whose initial part corresponds to this sequence is not feasible because it necessarily leads to a constraint violation.

In a Petri net representation precedence relations are qualitative and not quantitative as in AOA-graphs. The inconsistency expressed by the deadlock does not depend on the actual durations of the operations. Therefore it is a structural inconsistency.

In order to analyze how this kind of constraint violation can be pointed out, let us consider an example of a simple batch system.

A Example

Let us consider, a chemical process plant. The batches are quantities of fluid which are transferred from one equipment to another. Each transfer operation requires to allocate the destination equipment, transfer the material and, only then, to release the source equipment.

The plant is composed of an input buffer (IB), two reactors (Ra, Rb), an intermediate buffer (B) and an output buffer (OB) (see figure 1).

![Figure 1: Description of the plant](image)

The input and output buffers have an infinite capacity. Ra and Rb are identical, they are batch reactors (i.e. after a finite quantity of material is loaded, the reaction is executed). The capacity of buffer B is of one batch.

The batch process only produces one kind of product. The recipe is described by figure 2.

After each transfer, the reactor or the buffer is released, and can be use to another operation. The objective is to produce four batches.

B Petri net based characterization

The Petri net model is represented in figure 3, excepted the dotted arcs and place. It represents the set of constraints resulting from the precedence relations of the recipe (each operation is attached to a transition) and of the resource allocations. There is no quantitative information, i.e. operation durations are not explicit. In addition, the demand is not specified i.e. the net is cyclic and can accept any number of batches in any time horizon. The fact that we have put four tokens in place \( P_0 \) just denotes the fact that four batches will be allowed simultaneously in the process.

Let us remark that the firing sequence \( t_1, t_2, t_3, t_4, t_1, t_2, \)
\[ t_1, t_2 \] leads to a deadlock. It comes from the fact that the Petri net contains a siphon, and that this siphon can be emptied [13].

Let us remind that a siphon of a net \( N \ (\prec P,T,F \succ \) where \( P \) is a set of places, \( T \) a set of transitions and \( F \) the incidence matrix) is a set of places \( S \subseteq P \) such that 
\[ S \subseteq S^* \] (i.e. an input transition of a place of the siphon is also an output transition of a place of the siphon). When all the places of a siphon are emptied, it is not possible to fire any transition likely to increase its token load. And all the transitions having a place of the siphon as input are dead for ever.

The minimal siphon which is responsible for the deadlock in this Petri net is represented by a thick line in figure 3 (The set \( S \) comprises the places \( R, B, P_5 \) and \( P_6 \)). When \( t_5 \) is fired two tokens are removed from places \( B \) and \( P_5 \) and are put in places \( R \) and \( P_6 \). The number of marks of the siphon remains constant. But when \( t_1 \) is fired, the token count of the siphon is decreased by one (Three tokens are removed from places \( P_6, IB \) and \( R \), and two tokens are put in place \( P_1 \) and \( IB \)). And when transition \( t_5 \) is fired the token count of the siphon is increased by one (Two tokens are removed from place \( R \), included in the siphon, and \( P_4 \); and two are put in places \( B \) and \( P_5 \) which are both elements of the siphon).

Some approaches have been developed for deadlock avoidance by preventing the siphon to be emptied [1, 2, 6]. One of them consists in adding a place that controls the minimal numbers of tokens in the siphon. The dotted place in figure 3 is such a place. Its token load varies in the same way as the siphon token load and when it is empty the siphon token load is 1. We must verify that this place is not generating a new uncontrolled siphon which would introduce a new possibility of deadlock (it is not the case here).

With this place, the sequence leading to the deadlock can no longer be fired. When the sequence \( t_1, t_5, t_6, t_4, t_5, t_2 \) is fired, \( t_5 \) is not enabled, the only possible extension of the sequence is to fire \( t_5 \). Then \( t_1 \) can be fired again.

At this state every equipment is occupied, but the plant is not in a deadlock state.

By means of a reasoning over the Petri net representation of the structural constraints we have pointed out the possibility of deadlock and thus their inconsistency. By adding a new place, we have transformed the initial set of constraints into a new one which has the property that any sequence can infinitely be extended without any structural constraint violation. This property is very important for the supervision level. The plan can always be extended without violating the qualitative precedence constraints. Nevertheless we have no information about possible violation of quantitative constraints (i.e. the fact that, for example, some due date will not be respected).

C Activity-on-arc graph characterization

In this approach we explicitly operate on a time horizon corresponding to the production of exactly four batches. The AOA-graph describing the problem is represented in figure 4. It has an initial node (production start) and a
The recipe is represented by the horizontal paths. As there are four products to realize, there are four paths. The constraints resulting from the resources are represented by the other arcs describing precedence relations between operations on two different batches. Durations are normally attached to the arcs and good algorithms have been developed in order to derive the precedence relations verifying the resource constraints (mutual exclusion) and leading to the earliest due date.

In figure 4 the resources are allocated in order to represent the plan (the firing sequence) leading to the deadlock. The analysis of the structure of the graph shows us that the inconsistency of the plan is detected because a positive circuit in the graph (dotted arrows) appears. In the case considered here, as all the quantitative precedence constraints are operation durations, any cycle will be positive. This means that any legal plan is represented by an acyclic AOA-graph.

With this approach, the deadlock is also detected. However this detection is only valid for some specified demand and for some fully generated plan. The inconsistency will not be detected as a structural one and no general prevention approach can be developed.

4. DECISIONAL INCONSISTENCY

Let us now consider the example of two different recipes sharing the same machines. Recipe 1 consists in making an operation Op1 on machine M1 (1h) then Op2 on machine M2 (4h). And Recipe 2 consists in making two operations, the first one Op3 on M2 (2h) and the second one Op4 on M1 (3h). The objective is to make one product of each recipe.

The AOA-graph describing this problem is represented by figure 5. The two horizontal paths represent the two recipes. The two arcs labeled by the machine names M1 and M2 express the mutual exclusion constraint. They represent the fact that at least one of the two precedence relations has to be verified.

![Figure 5: Modeling of two recipes by AOA-graph](image)

In this representation the resources are not allocated to the operations. The four scheduling possibilities are represented (execute Op1 before Op4 or Op4 before Op1...). Two decisions have to be made, one for M1 and one for M2.

However they are not independent and for example if Op4 is selected first for machine M1 and Op2 is selected first for machine M2 a positive circuit will be generated (Op1, Op2, M2, Op3, Op4 and M1).

Once more this illustrates the fact that with AOA-graphs, it is not possible to guarantee the admissibility of a plan without generating it entirely.

B Petri net based characterization

Let us now construct the Petri net which would be obtained for the supervision of the same batch process. It is represented in figure 6. It is deadlock free and its firing sequences correspond to the three legal plans. Indeed, from the initial marking, transitions Op2 and Op4 are not enabled and so it is not possible to simultaneously select operation Op3 as the first one for M1 and Op2 for M2. As the Petri net is deadlock free, the resource allocations can be done in real-time, just taking into account the current state of the system, without any danger of obtaining a partial plan which could not be extended without violating some qualitative constraint.

![Figure 6: Modeling of two recipes by Petri net](image)
into an event graph by deleting some arcs which are inputs or outputs of places $M_1$ and $M_2$ and the non feasible plan will correspond to a non-live event graph.

It must be pointed out that structural and decisional inconsistencies will appear exactly in the same way in AOA-graphs: a positive circuit is generated independently of the actual durations of the operations. In contrast, with the Petri net approach for the supervision, the two situations are very different. In the first case, the qualitative constraints are potentially inconsistent and they have to be modified. In the second case, they are consistent and the inconsistent decisions are just avoided by the fact that only enabled transitions can be fired.

5. **QUANTITATIVE INCONSISTENCY**

Quantitative inconsistencies appear if a schedule cannot meet the time bounds (i.e., quantitative precedence constraints imposing a maximal delay between two operations or more generally between two events). Due dates for deliveries introduce this kind of constraints.

This case will be illustrated by the following example: make two products on the same machine with the constraints described in table 1.

### A Activity-on-arc graph characterization

The AOA-graph representation is composed of five nodes (figure 7).

![Figure 7: Modeling of a machine shop by AOA-graph](image)

The node $O$ corresponds to the beginning of the plan. The node $A_o$ denotes the beginning of operation $A$ and thus the delay between the beginning of the plan and that of operation $A$ is 3 (starting date). The label of the arc connecting $A_b$ to $A_e$ (end of $A$) is equal to the operation duration. The label of the arc connecting $A_e$ to $O$ is -9. It denotes the due date of $A$. Actually, if a path leading from $O$ to $A_e$ exceed the value 9, this would mean that the due date constraint is violated and in the AOA-graph a positive circuit would appear.

The two arcs $(B_e, A_b)$ and $(A_e, B_b)$ denote the mutual exclusion. Either $A$ is performed before $B$ or the contrary. If the decision is made to execute $A$ first then the plan is obtained by deleting the arc $(B_e, A_b)$. This AOA-graph $3 + 3 + 4 - 7 = 3$ pointing out the fact that the plan is not feasible.

Once more, the inconsistency is detected by a positive circuit. However, in this case, as some arcs have negative labels (maximal delay constraints) the fact that a circuit is positive or not depends on the operation durations and on the constraints values. It is not structural.

### B Petri net based characterization

The Petri net describing the recipes and the resource constraints is represented in figure 8.a. It can be used for supervision purposes. For scheduling, it can be used to simulate some sequence and verify that the due dates are respected or not. If we want to represent the decisions (execute $A$ before $B$) then it is necessary to transform the Petri net into an event graph and the net in figure 8.b is obtained. In such a graph it is known that the duration of the sequence between two nodes is obtained by considering the maximal length of all the paths connecting these nodes. For example, it can be seen that the length between $O$ and the end of $B$ is $\max(0 + 4, 3 + 3 + 4) = 10$ which exceeds the due date 7. Such an approach has been formally developed by [12] by means of his concept of dead marks.

![Figure 8: Modeling of a machine shop by Petri net](image)

However nothing avoid the combinatorial explosion of the paths analysis and the translation into a Petri net does not seem to be a help. Not surprisingly, when the inconsistency is the consequence of the decisions which have been made and when it depends on the actual values of durations, the structural analysis of the Petri net model is of little help. The information provided by the $p$-invariants and the $t$-invariants are only interesting when the issue is the elaboration of a plan with a cyclic behavior.

6. **CONCLUSION**

In this paper we compare a model commonly used for supervision (PN) with a model commonly used for scheduling (AOA-graphs). The objective was to compare these models with respect to their ability to detect that in a current state some decision can lead to a constraint violation or not.

AOA-graphs allow the detection of all infeasible sequences. But all the constraints violations are detected
Petri nets allow to detect infeasible sequences coming from the resource allocation constraints. The Petri net can be corrected to avoid these deadlocks. The second kind of inconsistency (decisional) will be automatically avoided by the fact that in a Petri net based supervision system only the enabled transitions will be candidate for possible decisions. The Petri net specificity (when it is used for supervision) is that it only depicts the constraints resulting from the recipes and the resource allocation mechanisms. The demand, the due dates and the plan are supplementary data allowing the supervisory control to decide which enabled transition will be fired next and at which date it will be fired. Not surprisingly, this approach does not offer specific tools to detect quantitative inconsistencies.

However, when the supervision level detects that the current state of the batch system is not the expected one, a rapid decision is required. Generally, it is more important to respect the recipe and resource constraints, and to guarantee that the plan can be extended until the completion of the product without reaching a deadlock state, than to exactly respect the due date. Indeed, due dates are constraints that can be relaxed whereas recipes and resources result in constraints which have to be respected in any cases.

The perspectives are to establish formal relations between the scheduling level and the supervision level. Two strategies can be outlined. The first one will consist in combining two models. One for the scheduling level (AOA-graph) and one for the supervision level (Petri net). The second will be to use one model (a Petri net or a AOA-graph) to implement both the scheduling level and the supervision level.

REFERENCES


