EVENT-BASED SUPERVISION OF A FERMENTATION PROCESS

Ramón Sarrate*, Julio Waissman*, Cherif Ben-Youssef* and Josep Aguilar-Martin**

*Universitat Politècnica de Catalunya, Edifici Vapor Sala, Rambla de Sant Nebridi, 10
08222 Terrassa, Catalonia
sarrate@esaii.upc.es

*Centro de Investigación en Tecnologías de Información y Sistemas, Universidad Autónoma del Estado de
julio@uaeh.reduaeh.mx
cherif@uaeh.reduaeh.mx

** Laboratoire d'Analyse et d'Architecture des Systemes, Toulouse, France
aguilar@laas.fr

Abstract: This paper describes an event-based approach to the design and implementation of a supervisory system applied to a biotechnological process. An heuristic model of the process under supervision is characterised as a discrete-event system (DES). Plant events generate transitions on the DES model, thus reflecting the dynamics of the continuous system underneath. This approach is applied to a yeast production batch process monitoring.

Keywords: Supervision, DES, Biotechnology.

1 Introduction

Nowadays, biochemical process industries are among the costumers on demand of an efficient supervisory system. Process supervision involves tasks such as control, monitoring or diagnosis of the plant. Situation assessment and fault prevention are of prime interest in order to increase process efficiency, afterwards, equates to benefit.

According to Kotch (1993), two main tasks are to be considered: process monitoring and supervisory control. Process monitoring comprises data collection and processing in order to have an updated plant state knowledge. At this stage the system must be able to decide whether the process is in an abnormal state and whether a corrective action should be undertaken. Such intervention along with diagnosis is regarded under the supervisory control task.

Biotechnological processes are particularly interesting for developing supervisory strategies due to the inexistence of a satisfactory mathematical model on which to implement classical model-based supervisory techniques. However, daily the operator interacts with these complex systems, identifying normal and abnormal situations and reacting accordingly. This fact shows that knowledge-based approaches seem appropriate to develop and design so-called intelligent supervisory systems (ISS). That is, systems which intend to implement human reasoning mechanisms in an automated way to carry out supervisory tasks.

A great deal of expert reasoning analysis is required in order to be able to abstract these mechanisms into the ISS. Many times, events are the main class of observations that experts are able to understand. For instance, as experts on physics, we know that after throwing a ball upwards it will return to the floor in a few seconds. Failure to do so will mean something abnormal has succeeded, which will make us look upwards in case someone has got it. The ball throwing, a timeout, the ball return, etc., are all events, upon which we reason and draw conclusions.

In fact event-based monitoring task involves two main intrinsic stages. On the one hand, the expert is
able to detect significative events from the process under supervision. Significative events carry useful information for the expert. On the other hand, he reasons based on these observations, carrying out his supervisory role. That is, monitoring and supervisory control. Following the previous example, a timeout can be indicative of an abnormal situation whereas an increase in ambient temperature has nothing to do with the ball not returning back.

This paper deals with the design and development of event-based supervisors based on a two layered hierarchical structure (Fig. 1). Under this scheme, in the event generation layer, process signals are continually monitored and processed in order to abstract events which are relevant to the reasoning layer. This second layer handles those events to draw conclusions about the state of the plant (i.e., process monitoring), with the possibility of reacting accordingly. Heuristic knowledge of the plant is used in order to construct both layers, as it will be shown in the application example.

![Diagram of Two层智能监督系统](image)

**Fig. 1. Two layer intelligent supervisory system scheme**

The reasoning layer must deal with events. According to the literature such a system is known as a discrete-event system (DES). Automata theory is used to express the DES model of the plant upon which to make decisions. Automata theory (Hopcroft and Ullman, 1979) is a convenient formalism to express DES models and to carry out supervisory control (Ramadge and Wonham, 1989).

These ideas has been applied to a fermentation process. Measured signals available from the fermentation reactor are analysed through the sliding-window methodology in order to obtain relevant trends. A DES model of the biotechnological plant is used to determine which batch stage is in progress.

2 Intelligent supervisory system design

In the development of an ISS, the hierarchical structure showed on Fig. 1 is to be followed. The ISS must be design according to the specifications stated by the expert. Based on this design goals, the engineer will have to decide which is the reasoning strategy most appropriate to achieve such goals, and which are the relevant events that better abstract useful information for the reasoning layer.

Such selection it is not so straightforward and the development of both layers should be done jointly. In fact, an interactive design process will be necessary according to the interactions showed on Fig. 2: reasoning will be carried out based on the events available, and specifications will accommodate feasible goals which the ISS can accomplish.

![Diagram of Main interactions in the ISS development](image)

**Fig. 2. Main interactions in the ISS development.**

In a second developing phase, the engineer will design the event generation layer using the sliding-window approach (Aguilar-Martin et al. 2003), and the event reasoning layer applying the DES formalism.

2.1 Event Generation Layer.

This layer is in charge of detecting relevant events from measured signals. In fact, building this layer is not an easy task that the engineer must confront. Dialog with the expert will be of prime importance, and a great deal of process knowledge and signal processing.

Upon the whole set of possible events $\Sigma$, the engineer must decide which are the relevant ones for the ISS, $\Sigma_R \subseteq \Sigma$. Relevant events will be transferred to the reasoning layer, so they must abstract useful information from plant signals. Such events can be associated to thresholds, trends, frequency content or temporal shapes.

Commonly, relevant events will be obtained by the combination of other events (not necessarily relevant) which will be generated from measured signals. So, the event generation layer can be subdivided in two sublayers: the generation of atomic events and the combination of those events (Fig. 3).

Under this approach, data is processed in sets comprising a time window. Each window is analyzed in order to produce an attribute which constitutes new
data. Applying this algorithm recursively with different kinds of analysis, the required events can be robustly generated. For instance, signal trends can be generated by conveniently chaining a regression analysis and a statistical analysis with histograms. This method was successfully applied to gas turbine supervision for process monitoring and situation assessment (Sarrate, et al., 1996).

Then a set of triangular symbols is defined which embeds slope and concaveness information about the input signal. Rengaswamy and Venkatasubramanian (1995) provided robust algorithms under noisy environments. Colomer et al. (2003), increased the triangular symbols by defining qualitative interval descriptions for the first and second derivative.

2.2 Event-Based Reasoning Layer.

This layer receives relevant events from the event generation layer and must assess the operator in their decisions, with possibly automatically reacting as needed. This paper just deals with the monitoring task, leaving the supervisory control task as future work.

The process under supervision is described by a logical DES model. It is logical in the sense that it includes no explicit timing information. It just expresses state sequencing based on relevant events observed in the plant. Again a Moore machine is used to formalise the plant model. Now the six-tuple is \((P, \Sigma_R, \Sigma_O, \phi, \mu, p_0)\), where:

- \(P\) is a set of states,
- \(\Sigma_R\) is the input alphabet,
- \(\Sigma_O\) is the output alphabet,
- \(\phi: P \times \Sigma_R \to P\) is the transition mapping which is allowed to be partially defined,
- \(\mu: P \to \Sigma_O\) is the output mapping, and
- \(p_0\) is the initial state.

The Moore machine helps in generating complex events from atomic events, with the latter being easier to detect. For instance, to detect a narrow pulse in a measured signal, a Moore machine \(M\) can be used, with \(\Sigma_R = \{\text{pulse\_detected}, \varepsilon\}\) and \(\Sigma_A = \{\text{signal\_increased}, \text{signal\_decreased}, \text{signal\_steady}\}\), like is shown in Fig. 4. So, whenever the input signal increases after decreasing, or decreases after increasing, \(M\) generates the event pulse\_detected.

States are associated to the output alphabet through the mapping \(\mu\). The output events, \(e_0 \in \Sigma_O\), are messages or information on the present process state that the operator (i.e., the expert) needs to know. For instance temperature\_alarm\_on\_thermocouple\_A. Every output event has a relevant meaning for the expert, not only as a diagnostic message, alarm or faulty situation, but also as a normal physical process.
state. In that sense, this Moore machine can be regarded as an observer of the process under supervision. The silent symbol must be considered as an output event with no relevant content for the expert.

The most difficult task is to define the transition mapping, $\phi$. As in the event generation layer design, the engineer needs a great knowledge of the process. Often, only through exhaustive dialog with the expert, the heuristic knowledge which he possesses on the process can be transferred by the engineer to this Moore machine. Transitions, governed by relevant events, express the dynamics of the plant. In that sense, relevant events change the process state. Such mechanism reflects the reasoning scheme followed by the expert. Waissman et al. (2003) propose a methodology to extract such expert knowledge using a Data Mining technique.

So, this Moore machine is used to describe a qualitative model of the process behaviour under normal, or even abnormal situations.

3 Application example: Yeast production

The supervisory architecture has been applied to a biotechnological process. S. cerevisiae is studied under oxidative regime to produce yeast under a laboratory environment in a fermenting reactor as shown in Fig. 5.

Two different procedures are applied: a batch procedure is followed by a continuous procedure. The batch procedure is composed by a sequence of biological stages. By this procedure the inoculum is introduced into the reactor and, progressively, biomass concentration (i.e., the yeast) increases to a maximum. This can be thought as an start-up procedure.

![Fig. 5. Fermentation plant.](image)

In the continuous procedure the substrate feeding input and the outlet are opened. Yeast is obtained by separating it from the substrate while the biomass concentration in the reactor is kept constant. It can be thought as a production procedure.

This paper deals with the batch procedure. Just a subset of two signals, which are available from the plant, are relevant: DOT and pH. Those signals will be processed through the sliding-window approach to detect trends as events. Reasoning is carried out on those trends in order to identify batch stages. According to the biotechnologist, four main stages compose this batch procedure, see Fig. 6:

i. The inoculum is introduced into the fermenting reactor along with glucose, which constitutes the substrate. During the first stage, biomass concentration increases, while glucose is consumed and ethanol produced. This stage is characterized by DOT decreasing.

ii. As soon as it has run out of glucose, there is an enzymatic transition. pH and DOT experiment a transitory dynamic.

iii. During the third stage, ethanol is consumed while biomass concentration increases. Again DOT decreases.

iv. Finally, as soon as it has run out of ethanol, there is a big transitory change. Normally, at this stage the biotechnologist initiates the continuous procedure. So this is the end of the batch procedure. This stage is characterized by a sudden increase in DOT and pH.

![Fig. 6. Batch procedure stages with (a) demand of oxygen transfer (DOT) and (b) pH. For (i) to (iv) see text below. Vertical axis scale is normalized.](image)
Two Moore machines are defined:

- MC, which generates $\Sigma_R$ from $\Sigma_A$. This machine composes (i.e., ‘+’ operation defined above) trend events and produces relevant events.
- MR, which generates $\Sigma_O$ from $\Sigma_R$. This machine reproduces the process behaviour, identifying the batch stages.

The whole ISS architecture has been implemented in SIMULINK. Data collected from the real plant is processed on-line by this ISS SIMULINK model. Both Moore machines, MC and MR, have been implemented through if-then rules. So the DES block constitutes the event combination sublayer along with the event-based reasoning layer of Fig. 3.

The results obtained are showed in Fig. 7. The ISS is able to correctly identify the four batch stages declared before. Every step change in (c) is univocally related to a process stage transition. There is a delay due to the causal scheme used in the event detection algorithms. Sliding-window algorithms have several parameters which have been manually tuned by checking the consistency of the results obtained running the whole system on a set of several scenarios available. So, the same ISS is able to correctly supervise the plant under different conditions, due to its robustness.

![Fig. 7. ISS output: (a) DOT, (b) pH and (c) the observed batch stages. Vertical axis scale is irrelevant.](image)

**Conclusion**

An efficient ISS architecture has been proposed in this paper. It continually monitors the plant detecting relevant events which are later used to observe the state of the plant under supervision. Automatical knowledge of the plant state is important for the operator job. This prevents human errors and increases the efficiency of the whole plant.

The ISS layered structure implements the sliding-window approach to detect atomic events from measured signals. Those events are processed by hierarchised Moore machines which intend to mimic expert reasoning schemes.

Future work will involve new applications in other domains which will allow the ISS architecture to progress towards supervisory control, extending and refining its present capabilities.

**References**


Colomer J, Meléndez J and Gamero, FI. Qualitative Representation of process trends for situation assessment based on cases,15th IFAC World Congress, Barcelona, 2002.


