New Perspectives in Industrial Control

Michel BERTRAND
ENSAM CER de Lille
8 Bd Louis XIV 59000 Lille - France
Fax: 33 03 20 53 55 93
e-mail: bertrand@lille.ensam.fr

Abstract
This paper aims at providing a brief overview of industrial control methods; it takes into account the evolution arised in the last years, and attempts to point up some perspectives. At first it lists the different steps of the life cycle of the Automated Production Equipment, then expounds the need for several types of models, corresponding to different aspects of the control development, and shows the problems brought up by their obtention or their validation. It reviews some control methods, with special emphasis on their anticipatory aspects. It tries to make clear not only the main ideas, but also the operating conditions corresponding to each method.

Keywords: Industrial Control - Modeling - Specification - Anticipation

1 Introduction

When automatic control has started out in the industry, it was characterized by the following features:

- A strong decomposition of the problem, so that one machine at once was equipped.
- A division between the binary systems (using relays, switches and so on) and the continuous ones, which control temperature, flow, level, etc. This was true for the methodology as well as the realization, and corresponds more or less to the division between manufacturing and process industries, but the frontiers are not precisely delimited and many installations mix the two categories.
- The analysis of the process was based upon single input, single output (SISO) models.

Various stratagems reduced the problems to a lot of such models:

- The control algorithms might be partially empirical (for example the famous criterion of Ziegler and Nichols).
- The control part which depended on binary variables was treated by several methods: state diagrams, chronological tables, which often proved to be inadequate in an industrial context because of the high number of cases to study.
- The safety devices were not included in the controlling unit but acted directly and independently on the machine.
- The production management was traduced into setpoints and logical orders by the operators.

The control objectives were in a similar manner limited:
- keeping a nominal setpoint;
- automatic production cycle and, if a failure occurs, emission of an alarm.

Today, automatic control deals with large and more and more complex systems. New features become apparent:
- Numerical processors are present eveywhere.
- As a consequence, models and algorithms may be far more complex.
- Continuous and binary signals are treated by the same controllers.
- The different operating modes are taken into account.
- For a new equipment, the design of the control starts as the same time as the design of the whole system.

The control objectives become for their part more and more ambitious: they require
- to insure a good running, including optimization of energy, of wearing..., but also starting and stopping procedures;
- to face up to possible failures with different levels of procedure;
- to instantaneously obey the orders of the production management sent via an idustrial net, to provide real time data to monitoring , etc.

We shall deal here with some methods with good prospects, that we shall replace in an appropriate framework, with emphasis on the anticipative aspects of the control strategies. We do not show some practical application, but makes reference to the constraints of the operating conditions in different types of applications. We hope so to provide a good picture of the domain under consideration, even if it is far from an exhaustive review.

2 The Successive Steps

2.1 Life Cycle of an Automated Equipment

The story of a large automated system, from the first studies to the destruction, may today be defined by a life cycle (Pecht,1994). The typical cycle is represented in figure 1, according to a V-shape.

Other graphical representations have been proposed (spiral, "cascade"), but, for the most used ones, there is a general agreement on the main steps.
  - specification
  - conception
  - realization (including implementation of the control algorithms)
  - test
  - maintenance

and on the simultaneous elaboration, when it is possible, of the control and of the system itself.

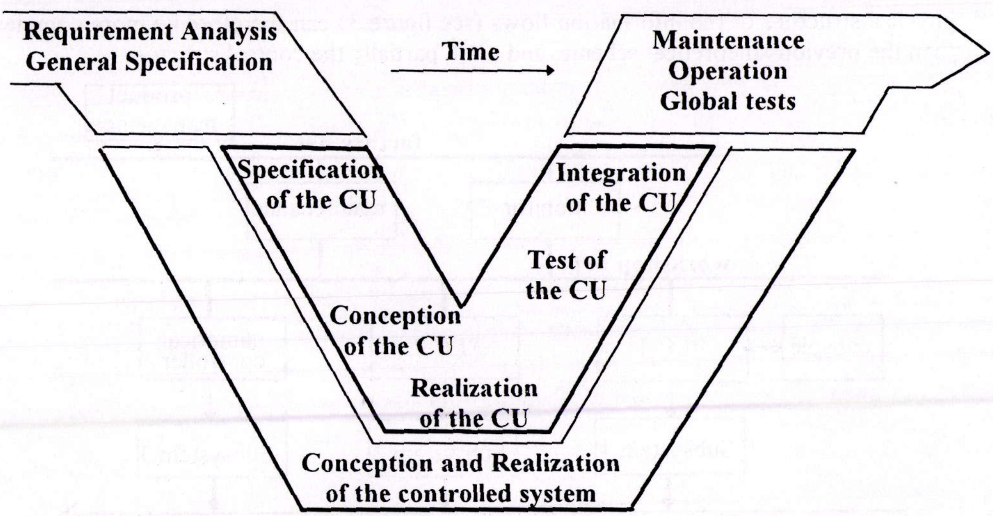


Figure 1: Life cycle of an automated production equipment

2.2 Architecture of the Automated System

Some important points must here be noticed:

- The controlling unit (CU), which is the heart of the system, is separable from the controlled system, even if they have been designed at the same time and in an interactive manner. The functional structure is then illustrated by the following scheme.

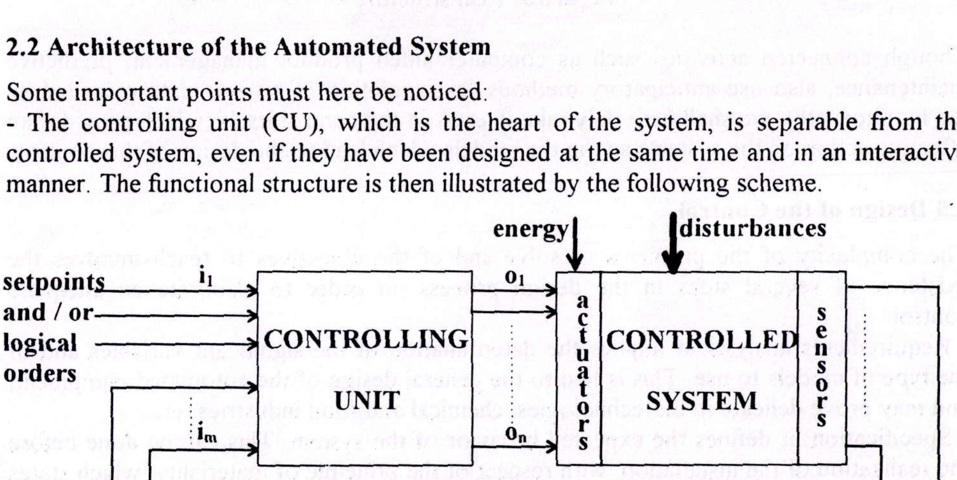


Figure 2: Functional structure

- The "world is seen" from the CU; inputs and outputs are defined according to this point of view.

- Other sectors of the plant (production management, maintenance) also use mathematical techniques, more and more sophisticated, needing real-time data. The physical structure of the information flows (see figure 3) can therefore be more complex than the previous theoretical scheme, and mask partially the control structure.

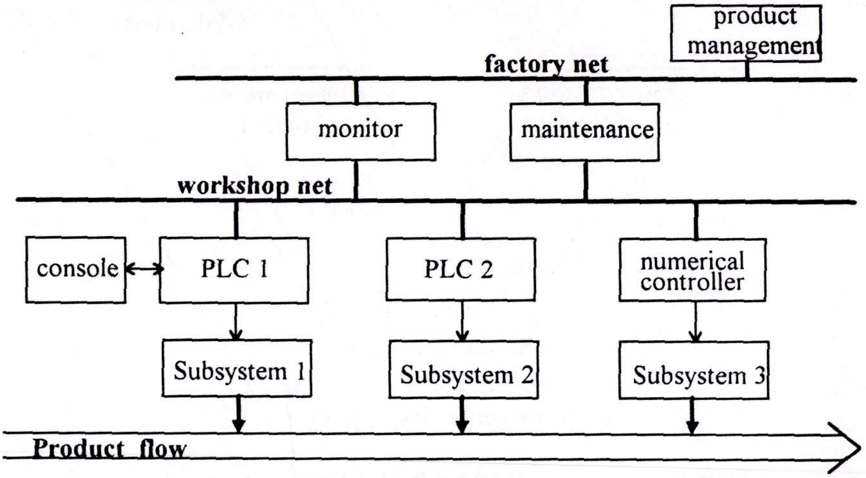


Figure 3: Real structure

Though connected activities such as computer aided product management, predictive maintenance, also use anticipatory methods (extrapolation of historic data recorded by the monitor, etc), we shall here only take interest in the control itself (which provides an effective action on the system) and in the models which lead to the design of this control.

2.3 Design of the Control

The complexity of the problems to solve and of the objectives to reach involves the existence of several steps in the design process, in order to elaborate an adequate control:

- Requirements analysis: it implies the determination of the significant variables and of the type of models to use. This is tied to the general design of the automated equipment, and may prove delicate in biotechnologies, chemical and food industries, etc.
- Specification: it defines the expected behavior of the system. This can be done before the realization of the installation, with respect of the principle of materiality, which states that if the system does not yet exist, it is intended to exist (and therefore has to respect the physical reality). At the end of this step, the general structure of the controlling unit must be defined.
- Conception: it results in a lot of control laws, elaborated with the help of control models.
    * Control models have to justify the dynamic behavior of the controlled system, defined by the signals provided by the sensors and the detectors, when the exterior exert actions on it. For many systems, these models can only be obtained after experiments, that is to say when the controlled system is built.

* Control laws translate the control objectives into control algorithms according to strategies based on the previously defined models

- Realization: the implementation of the control algorithms sometimes show that the computer behavior has to be carefully investigated.
- Validation of the control: this step is often neglected in the methodologic papers, but is very important: it includes the validation of the controlling unit itself, the integration controlling unit / controlled system and finally the tests of the whole system.

One can notice that the maintenance does not amount to the repair of the failures which may occur; it also includes modifications of the control. This requires a modular structure of the control and has to be taken into account in the conception step.

3 The Models

For control engineering, the model is often the heart of the matter Requirements for safety and quality make it indispensable. But a model is not an end in itself.

3.1 Specification Models

The different modes of operation are introduced in this part of the conception work. The states of the system are at first described globally, and then more and more precisely refined. In a first time, they appear as a lot of blocks associated to the main functions in a very general description. The expression of the role of each block is litteral, and so is the expression of the evolution conditions, which belong to the on / off type, such as the switch turned by an operator. Even if this fact is hidden since the supervised data are not binary ones (number of pieces, thresholds, equalities between several values...), each condition results in the response "True or False" to a question, or in a combination of such responses.

A formal interpretation can be given by the statement "a discrete events system supervises a continuous time system". Let us recall that Discrete Events Dynamic Systems (DEDS) use variables which, in a first approach, have only 2 values. Mathematically they are treated by the help of Boole algebra plus an event's one (Frachet, 1997, 1999) to take into account the physical time.

It is important to notice that different decompositions can be performed. There is no absolute method to establish a decomposition. Technical considerations interfere with purely functional rules. An usual approach consists in structuring the specification according to a hierarchical structure, the safety procedures having the highest priority. Another point is the necessity that the specification results in an operational control structure, and not only in an understanding or simulation tool.

The methodological tools used for this purpose are now given.

- The GEMMA (ADEPA, 1983) defines 16 blocks a priori corresponding to 3 major modes (operating, stopping, failure treatment). From a formal point of view, it is a state diagram, the present situation corresponding always to one block only. It suffers from its single level of representation and its rigidity. But it is very helpful to review all the possible cases, and, as the author has personnally verified, to define the communication tools with the operator. It is a good first analysis.

- Petri nets (David, 1992), with several extensions (timed, coloured or batch Petri Nets, for example). Their use provides a more complete description than the previous method, and allows formal verifications which are very important for large systems, such as

  * all the states (places) can be reached;
  * no locking can occur.

However, the representation may become too complex and difficult to structure in a hierarchical manner.

- The GRAFCET (David, 1995; Frachet, 1999): elaborated by a French association gathering industrial and academic partners at the end of the seventies, it benefits from 3 advantages

  * the possible modeling of simultaneous evolutions (parallel sequences, see fig. 5);
  * the standardisation (its principles provided the bases of the IEC 848 standard);
  * the easy implementation: there is a possible automatic translation into a programmable logic controller (PLC) program; in that case, it becomes indirectly a control tool .

The GRAFCET is therefore a global method for the control of DEDS; for the hierarchical structuration of specifications, some extensions, which are not (yet?) standardised, are desirable (Duméry, 1996): so an enclosure procedure would allow a more and more detailed representation of the behavior, with respect to decomposition rules.

Other specification models may be used, but, except if they do not consider the dynamic behavior, which is very restrictive, they present similarities with the previous ones: it is the case of the Statecharts (Harel, 1987), which introduce the notions of hierarchy and history, and of Grafchart, which is issued from the GRAFCET and the Statecharts.

Two approaches of a real example, a pilot drying installation, are illustrated in fig. 4. The system is of course only partly shown.

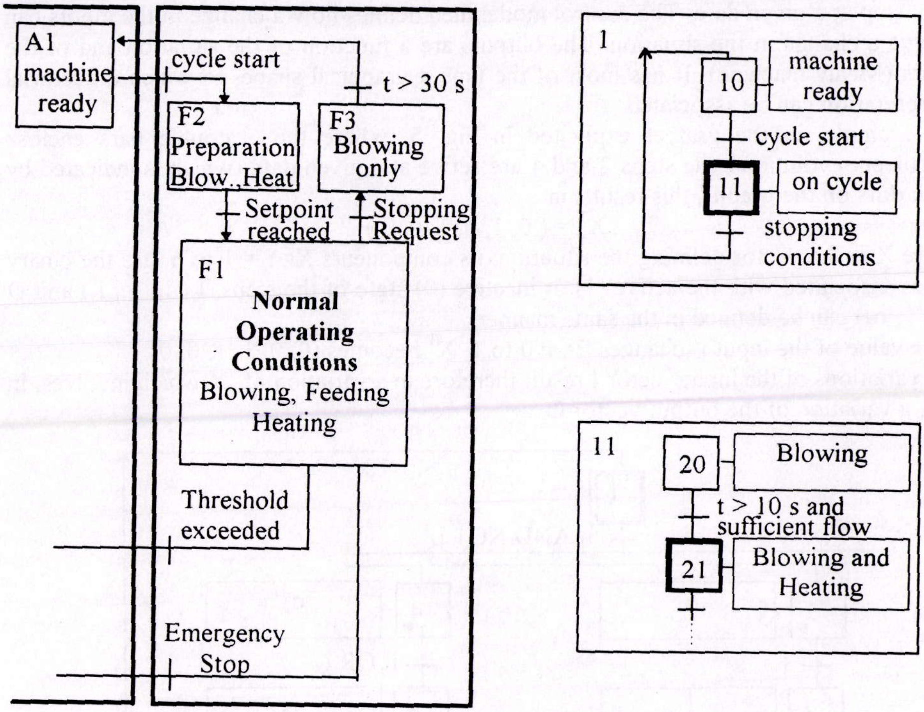
Fig. 4(a) corresponds to a GEMMA-type specification. The 3 blocks corresponding to operating modes, F1, F2, F3 are represented. The complete GEMMA defines a sort of grid, the most complex blocks of which must then be explicited by GRAFCET. There is unfortunately no much help to perform this work, and the result requires a reduction before implementation: for example, "blowing" is present in F1, F2, F3.

Fig 4(b) corresponds to an extended GRAFCET. It shows the first 2 levels of the embedded systems (4 levels seem to be a maximum in industrial cases). The method provides a structured decomposition which is easily translated into a control algorithm.

The difficulties arise from:

* The definition of the steps, that is to say the objectives at each level; notice that the definitions at level 1 are very wide (see here the step 11), and must be precised (an enclosed GRAFCET explicits the "hyperstep" 11, and the step 21 of this chart will itself be precised at a 3rd level).

* The rules of decomposition, which must avoid contradictory orders; the use of a methodology issued from the Statecharts and of the organic structure of the real system itself may help to obtain a clear tree diagram.



(a) **(b)**

Figure 4: Specification of a pilot dryer

An important problem is the verification of the specification models. One often tries to insure the absence of failures, and, more generally, of unusual situations. The policy then consists in simulating the different scenarii to detect possible failures. Unfortunately, in many real problems, the number of situations becomes very large and a reduction must be operated through two approaches:

- by decreasing this number thanks to specific information resulting from the structure itself or from surrounding factors;
- by testing a lot of situations, according to a random choice, to insure a high probability of satisfactory responses.

3.2 Control Models

They must be able to predict the behavior of the system under a known external action. This is true in the continuous as well as in the discrete area.

3.2.1 DEDS Models

The modern methods of modeling describe the behavior by the mean of a set of steps (places). The physical structure is shown Fig. 2. The situation defines the binary value of each step at a given date. The control model then defines how a change in the inputs can involve a change in the situation. The outputs are a function of the situation and of the inputs (Mealy machine). It has most of the time a graphical shape, to which a vectorial interpretation can be associated.

For example, in the grafcet explicited in Fig. 5, where the 2 double bars enclose concurrent evolutions, the steps 2 and 4 are active at a given date (which is indicated by black dots on the graph); this results in

$$X^T = (0, 1, 0, 1, 0, 0)$$

where X is the vector defining the situation; its components Xi, i = 1 to 6, are the binary values associated with the active (1) or inactive (0) state of the steps. I ($i_1, ..., i_5$) and O ($o_1, ..., o_4$) can be defined in the same manner.

If the value of the input i_2 changes from 0 to 1, X^T becomes (0, 0, 1, 1, 0, 0).

The variations of the input vector I result therefore in a variation of X which involves, in turn, a variation of the output vector O.

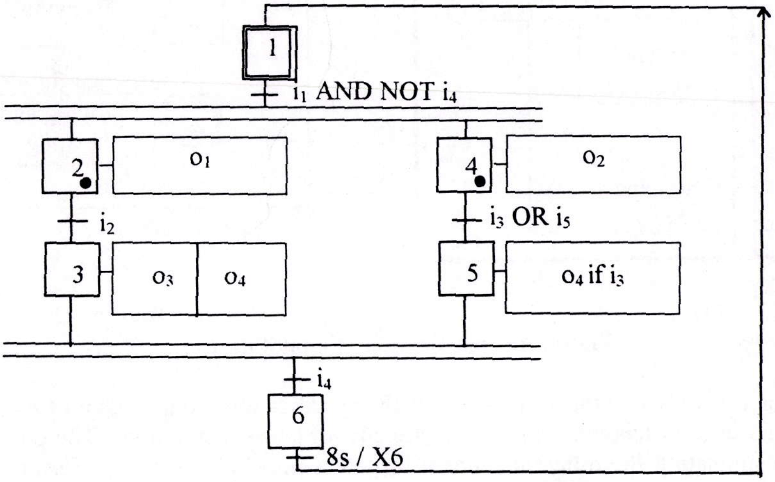


Figure 5 Example of discrete control model

This modeling results in a chronology but industrial requirements have involved the creation of time dependant variables (8s / X6 in the previous figure), which can provide a binary value corresponding to a duration. Most of the time this is done in the control unit by counting the pulses of a clock, with frequential division procedures.

Notice that the use of the methods proposed in the § 3.1 makes the obtention of such control models easier, those models being only developments of the first description, with the same tools, but the inputs and outputs are the physical signals.

3.2.2 Continuous Behavior

Except for manufacturing systems which only depend on binary signals, most of industrial systems include continuous variables (temperature, level, pressure, moisture...).

The dynamic behavior of these analog outputs under the stimuli of inputs such as the signals applied to the actuators (valves, motors, cylinders,...) is represented by control models, most of the time by the way of a set of differential or recurrence equations.
Two procedures provide such models:

- physical analysis according to relations expressing the preserving of flow, energy, force, etc.; the equations reveal often complex, non linear, and the numerical values of the different coefficients are difficult to estimate;
- identification (Ljung, 1987) from experiments on the system; it is then necessary
  * to choose a significant (from an informational point of view) input signal (stimulus) that the process must be able to endure;
  * to choose the structure of the model (characterization);
  * to determine the parameters of the chosen structure.

The second point is still semi-empirical. For the third one, various methods have been proposed. The least square methods and its numerous extensions are very popular, but it is not always easy to eliminate the bias which can reveal.
A present tendancy consists in identifying the process according to a predetermined structure, which can result from a qualitative physical analysis.
The behavior of the process faced with measurable disturbances (such as the outside temperature in thermal systems) is obtained in a similar way.

4 The Control Laws

The DEDS and the continuous or digitized ones are classically governed by separate control algorithms, the expression of which, as for the control models, is quite different. This looks today like a paradox, since the control organ is the same or belongs to the same type in the two cases. We shall investigate the two types before considering a possible association.

4.1 Discrete Systems Control

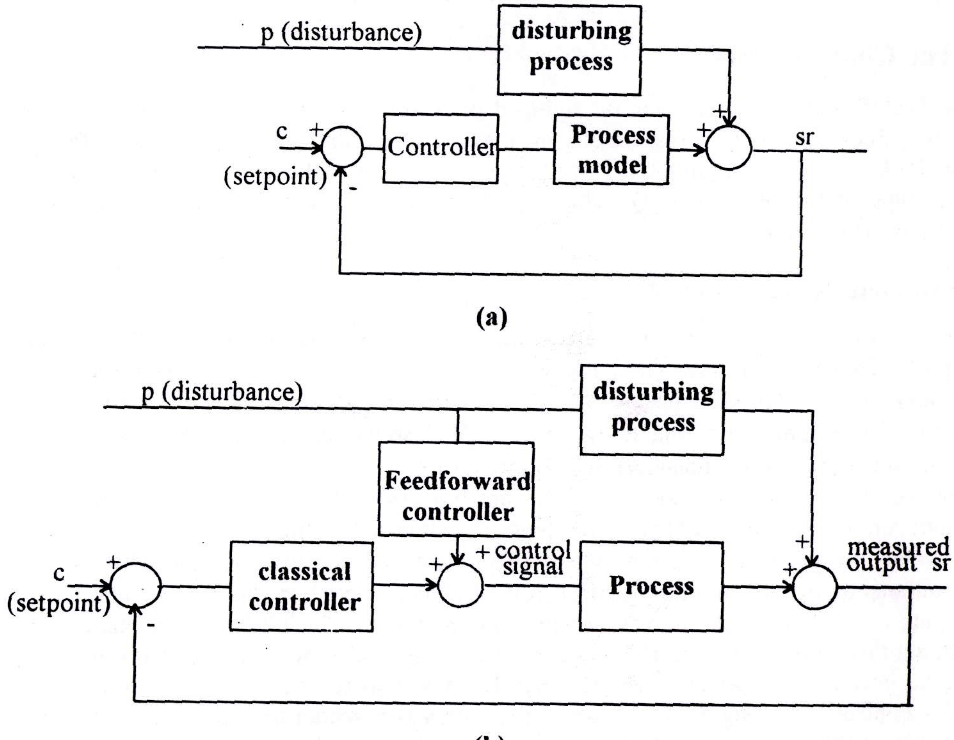
This part can be seen as easy when the control model is established (cf §3.2.1). To each step of a GRAFCET is associated a boolean function generally explicited as a Set-Reset memory. If the GRAFCET has been used, many PLC's convert automatically a graph in a control program, according to the IEC 1131-3 standard, in which the "Sequential Function Charts " are defined as a structuration tool.
However some problems may occur: they are principally due to the fact the controller cannot run infinitely quicker than the exterior, as supposed by the model.
Whatever the used method, this temporal problem and its corollary, the existence of "simultaneous events" is one practical difficulty encountered in the implementation of discrete control. To make clear this point, the model must include the controlling unit, with another time scale. Controversy arises from the synchronous or asynchronous way to take into account the events translating the impact of the outer word on the system under control. A strong structuration of the model is essential to make the description readable and the maintenance easier; sometimes it also provides the distribution between several control processors (see fig. 3).

4.2 Continuous Systems Control

The control of continuous systems is generally performed by algorithms based upon models using linear transformations (Laplace, Fourier) of continuous functions, or upon recurrence equations, directly suitable for the numerical processors. The control objectives result in requirements about stability, static or dynamic precision, transient response, when the setpoint is modified or a disturbance occurs, or, in other terms, precision, dynamics and robustness, this last word expressing the keeping of the main properties, and at first the stability, in case of a poor modeling or a change in the system itself. The Proportional Integral Derivative (PID) algorithm remains, with some improvements, the industrial reference in this area. But its inadequacy in some cases (important delays, integrator or instable processes, varied constraints), has involved the designing of other algorithms, several of which present anticipative aspects, which will be precised in each case. We shall here examine the most significant algorithms.

4.2.1 Feedforward Control

Its anticipative aspect appears if one compares the response of the control structure described in Fig. 6(b) with the response of the classical loop (Fig. 6(a)).



(a)

(b)

Figure 6: Feedforward control (SISO case)

In the simplest (but very used) case when the process can be represented by a linear model, we can write for the classical loop, according to the formalism based on the Laplace transformation

$$sr(s) = (c(s)\, A(s)\, C(s) + p(s)\, G(s)) / (1 + A(s)\, C(s)) \qquad (1)$$

A(s) being the model of the process, G(s) the disturbing process and C(s) the algorithm of the controller.

The principle of superposition allows to study separately the effects of p and c. So, if FC(s) is the feedforward control algorithm (Fig. 6(b)), the behaviour when the disturbance p varies obeys to the relation

$$sr(s) = p(s)\, (G(s) + A(s)\, FC(s)) / (1 + A(s)\, C(s)) \qquad (2)$$

Ideally one aims at cancelling the effects of p; the perfect feedforward control anticipates those effects to fight them, and verifies

$$G(s) + A(s)\, FC(s) = 0 \qquad \text{or} \qquad FC(s) = - G(s)\, [A(s)]^{-1} \qquad (3)$$

Most of the time the inversion of A(s) cannot be achieved. FC is ordinarily only Proportional (static anticipation) or Proportional and Derivative; it allows a decrease of the maximal error due to p and an acceleration of the response.

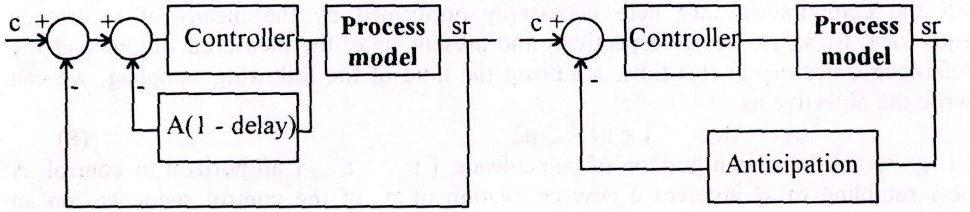
Feedforward control is an open loop control, so it does not replace the feedback control, which insures the response to the setpoint changes and the effects of non measured disturbances, but adds itself to this control, as it appears in Fig 6(b).

Notice that it is a reduction of a multivariable system, in which the disturbance p would be considered as an ordinary input.

Feedforward control offers potential improvements, and can be applied in the multiple input - multiple output (MIMO) case (Schnitzlein, 1994).

4.2.2 Smith Predictor

It is used when the system presents a considerable delay. Its role consists in getting the delay out of the loop, as if the feedback did not take the delay into account. Its name is justified by the equivalence, from a mathematical point of view, between the schemes (a) and (b) in Fig. 7. This equivalence is somewhat paradoxical, since (a) contains a delay and (b) a pure anticipation, which is of course unachievable.



(a) **(b)**

Figure 7: Smith Predictor

According to the formalism defined in 4.2.1, the model of the process is now $A(s)\, e^{-Rs}$. The tracking behavior (when c varies) with a classical control is then described by (4)

$$sr(s) = c(s)\, A(s)\, C(s)\, e^{-Rs} / (1 + A(s)\, C(s)\, e^{-Rs}) \qquad (4)$$

The use of an anticipation e^{Rs} in the feedback (Fig 7(b)) leads to (5)

$$sr(s) = c(s)\ A(s)\ C(s)\ e^{-Rs} / (1 + A(s)\ C(s)) = c(s)\ B(s)\ e^{-Rs} \qquad (5)$$

where B(s) contains only polynomials, which allows to apply classical rules in order to elaborate C(s) from a previously fixed B(s): for example B(s) is a second order system with a given damping ratio.

Many systems, such as the temperature evolution in the example given above (fig. 4), can be modelized by a delay and a first order equation

$$A(s)\ e^{-Rs} = K\ e^{-Rs} / (1 + \tau s) \quad K, R, \tau \in R^+ \qquad (6)$$

The objective is then

$$B(s) = 1 / (1 + \tau' s) \qquad \text{with } \tau' < \tau \qquad (7)$$

The "compensation" of the delay in this case is performed by industrial preprogrammed controllers. The corresponding algorithms are more robust (as explained at the beginning of § 4.2) than the classical ones. This is also true for the response to disturbances (which are not represented in Fig. 7).

4.2.3 Model Predictive Control (Richalet, 1978)

It has appeared at the end of the seventies and uses directly the potentialities of the numerical controllers. It is an improvement of the Internal Model Control, by the introduction of an anticipative strategy.

The main features are the following ones:

- An internal model of the process, that is to say a model which is implanted in the computer and explicitely used in the algorithm. All types of models are possible, even those expressed by rules as in fuzzy logic.
- A reference trajectory - often a first order system - which takes into account the impossibility of an instantaneous response. This trajectory is at each sampling instant reinitialised on the measured output(s).
- A control strategy including the existence of constraints on the control signal(s) or on the controlled outputs (threshold, maximum value for the gradient...)

The basic procedure consists then in computing an output sequence in order to insure the coïncidence between the model response and the reference trajectory on a prescribed future horizon, and to apply the first signal(s) so determined to the real system.

All the computations are here necessarily performed by the means of recurrence equations. t_k, s_k, rt_k being respectively the present time, the measured output and the reference trajectory at this time, t_{k+1} being the time of the following sampling, we can write the objective as

$$sr_{k+i} = rt_{k+i} \qquad i = n1, \ldots, n2 \qquad (8)$$

$[\ t_{k+n1}, t_{k+n2}\ ]$ defines an horizon of coïncidence, $[\ t_{k+1}, t_{k+n2}\ ]$ an horizon of control. A new sampling of sr involves a new calculation of rt, of the control sequence, on an horizon shifted forward: $[\ t_{k+2}, t_{k+n2+1}\ ]$; this is known as the principle of sliding horizon.

In order to determine the control sequence, as for the modeling, many methods can be carried into effect. Simulation may turn out to be a helpful tool.

Some improvements may be added to this first approach:

* A "self-compensation" by adding to the computed output a function of the difference between the process and model outputs (sr - sm, in Fig. 8) . This allows,

by working in the frequential domain, to fight bias for example at low frequencies. Notice that the static precision results from the internal model structure itself.

* Introduction of the influence of the measured disturbances, as in the previously described feedforward control.

* Adaptation of the internal model (Diaz, 1996) illustrated by dotted lines in Fig. 8.

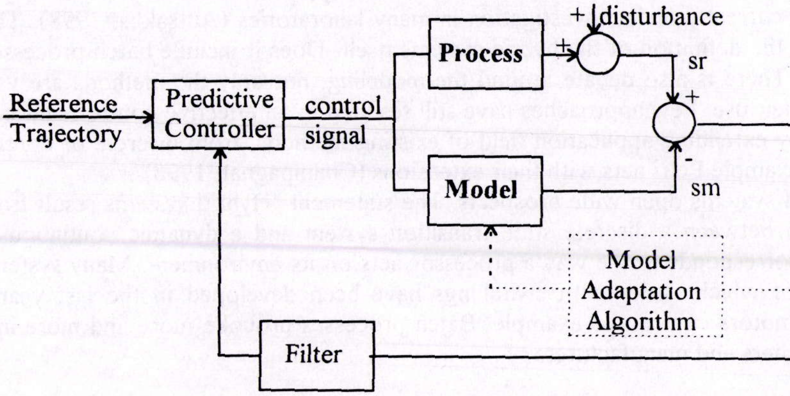


Figure 8: Model Predictive Control (SISO case)

Contrary to the two previous methods, which are only accelerating or simplifying procedures, MPC uses a true anticipation strategy. There is an explicit feedback from a (computed) future toward the applied control signal.

4.2.4 Other Control Algorithms

Other techniques have been developed, overriding the control problems. We will only mention fuzzy logic and neuronal networks, which play an increasing role in modeling and behavior prediction for systems which offer poor information, uncertainty, or badly structured knowledge. Different control applications have been successfully performed, and even anticipative strategies elaborated with the help of those tools (Perrin, 1996): a fuzzy algorithm is used in the control structure of a chemical reactor to predict the influence of a delay.

But if they are popular in the robotics area, or in pattern recognition, the applications in industrial control seem to be rather limited. Most of the fuzzy controllers are variants of the Proportional Derivative type. They increase the robustness in comparison with a classical controller, but must be completed by an integrator to insure the precision.

Incursive anticipatory control (Dubois, 1998a, 1998b), non linear algebra, also offer possibilities to solve complex algorithms. Their exploration in the considered area is at its outset. Hyperfinite signal provides a representation of a sampled signal which has only a finite number of predefinite values. If the cardinal of this set of values is p^n (p prime number, n integer), it has a field structure with interesting properties. This is true for most of the industrial systems, because of the digitization ($p = 2$). The possible use for DEDS ($n = 1$) has already been proved (Frachet, 1997). Present research tries to extend the infered algebra to classical sampled data systems, in an unifying approach which is also the aim of the hybrid systems control.

4.3 Hybrid systems control

The junction of the discrete and the continuous approaches leads to hybrid systems, which are currently under investigation in many laboratories (Antsaklis, 1998). There is debate on the definition of the hybrid systems itself. Does it include batch processes, for example? There is also debate around the modeling: not only the methods are various, but also their use. Few approaches have still resulted in an effective control, and most of the time by extending application field of existing methods, from discrete or continuous area, for example Petri nets with their extensions (Champagnat, 1998).

But hybrid systems open wide prospects. The statement "Hybrid systems result from the interaction between a discrete state transition system and a dynamic continuous one" basically corresponds to the way a processor acts on its environment. Many systems the behavior of which varies with switchings have been developed in the last years, for electrical motors control for example. Batch processes provoke more and more interest from searchers and manufacturers.

5 Conclusion

The first assessment to be highlighted is the universal use of computers. This considerably enlarges the control abilities: numerical processors can achieve more elaborate algorithms than the PID one or the set/reset memory. Simulation, at different levels, is become a fully recognized tool.

Various control methods are issued from this evolution. The most performing ones, in the discrete as well as the continuous area, attempt to obtain a specified behavior. We have insisted here on this point, which is essential and not always easy to satisfy in a real context. However, the multiplicity of the proposed methods, issued from quite different approaches, require integration procedures to recover the results of previous steps of the design procedure. The documentation problems, the technical difficulties to transfer data by nets between processors, the misunderstanding of a too complex reality sometimes conceal the importance of the methodology from the control engineers (Galara, 1997).

Modern control is often able to strongly reduce the difference between the objectives and the obtained results, even in the case of disturbances or failures, why they are provided in the control policy, as illustrated here on some simple cases.

Without neglecting the theoretical and practical impediments which restrain the development of new methods, we can assert that industrial control will more widely use a prediction of the behavior in various cases. Theory brings new tools which offer new perspectives, even if they require cautious validation.

References

ADEPA (1983). GEMMA, Guide d'Etude des Modes de Marche et d'Arrêt, (in French), Association pour le Développement de la Production Automatisée, Montrouge (France).

Antsaklis Panos, Koutsoukos Xenofon, Janan Zaytoon (1998). On Hybrid Control of Complex Systems: a Survey, European Journal of Automation (APII-JESA), Vol. 32, n° 9-10, Editor Janan Zaytoon, Hermès Science Publication, pp 1023-1045.

Bertrand Michel (1998). Le controle/commande industriel: une évolution illustrée sur un cas concret (in French), Proceedings of the 15th International Congress of Cybernetics, published by the International Association for Cybernetics, Namur (Belgium), pp 276-282.

Champagnat Ronan, et al. (1998). A gas storage example as a benchmark for hybrid modelling: a comparative study, Proceedings of the 3rd International Conference on Automation of Mixed Processes (ADPM'98), Editor J. Zaytoon, publisher Presses Universitaires de Reims (France), pp 123-130.

David René, Alla Hassan (1992). Petri nets and Grafcet - Tools for modeling discrete events systems, Ed. Prentice Hall, London.

David René (1995). GRAFCET: a Powerful Tool for Specification of Logic Controllers, IEEE Transactions on Control Systems Technology, Vol. 3, n°3, pp 253-268.

Diaz Constantino, et al. (1996). Simultaneous adaptive predictive control of the partial pressures of dissolved oxygen and dissolved carbon dioxide in a laboratory-scale bioreactor, Journal of Biotechnology, 52, pp 135-150.

Dubois Daniel M. (1998a). Incursive Anticipatory Control of a Chaotic Robot Arm, Computing Anticipatory Systems: CASYS'97, ed. by D.M. Dubois, American Institute of Physics Conference Proceedings, Woodbury, New York, 437, pp 406-417.

Dubois Daniel M. (1998b). Modelling and Simulation of Hyperincursive Anticipatory Oscillators, Proceedings of the 15th International Congress of Cybernetics, published by the International Association for Cybernetics, Namur (Belgium), pp 348-353.

Duméry Jean-Jacques, et al. (1996). A tool for the structured modeling of discrete events systems behavior: the HyperGrafcet, IEEE / IMACS Multiconference Computational Engineering in Systems Applications, Lille (France), pp 519-524.

Frachet Jean-Paul (1998). The Hyperfinite Signal: A New Concept for Modelling Dynamic Systems, Computing Anticipatory Systems: CASYS'97, edited by D.M. Dubois, American Institute of Physics Conference Proceedings, Woodbury, New York, 437, pp 495-506.

Frachet Jean-Paul, Bertrand Michel (1999). GRAFCET-SFC (Sequential Function Chart) and its extensions as a formal base for modeling the behavior of complex industrial systems, SCI'99 / ISAS'99, Orlando, ed. by Nagib Callaos, published by tje International Institute of Informatics and Systemics, Stokie, Illinois.

Galara Dominique, Hennebicq Jean-Pierre (1998). The Gap from Scientific Systems Approach down to Industrial Power Plant Control Engineering, International Journal of Computing Anticipatory Systems, vol. 2, pp 50-77.

Harel D. (1987). Statecharts: a visual formalism for complex systems, Science of Computer Programming (Netherlands), vol. 8, n° 3, pp 231-274.

Ljung Lennart (1987). System Identification. Theory for the user, Ed. Prentice Hall, Upper Saddle River, New Jersey.

Pecht M., Lall P. (1996). A Life-Cycle Approach to Design and Manufacturing Simulation of Multichip Modules, International Journal of Modelling and Simulation, vol. 16, n° 4, pp 173-183.

Perrin Eric, et al. (1996). Nouvelle stratégie simplifiée de commande floue d'un bioréacteur continu à levures immobilisées (in French), LFA'96, Cepadues Editions, Toulouse (France), pp 186-193.

Richalet Jacques, et al. (1978). Model predictive heuristic control: applications to industrial processes, Automatica, 14,(5), pp 413-428

Schnitzlein Klaus (1994). A Learning Feedforward Control Strategy for the Concentration Control of Recycle Reactors, IASTED 13th Conference on Modelling, Identification and Control, Grindelwald (Switzerland), ed. by M. H. Hamza, published by the International Association of Science and Technology for Development, Calgary, Canada.

Zouari B., Taghelit M. (1996). An Interpreted Coloured Petri Net Model for Verifying Protocol Specifications, International Journal of Modelling and Simulation, vol. 16, n° 2, pp 49-58.