Supervision and Advanced Control in a Beet Sugar Factory

Authors: C. Prada (*), C. Alonso (**), F. Morilla (***), M. Bollain (****)

(*)   Dpt. Systems Engineering and Automatic Control, Faculty of Sciences,
          University of Valladolid, 47011 Valladolid, Spain.
          TL . + 34 983 423164, Fax + 34 983 423161, e-mail prada@autom.uva.es
(**)   Dpt. Computing Science, Faculty of Sciences,University of Valladolid,
          47011 Valladolid, Spain.
          TL . + 34 983 425005, Fax + 34 983 423161, e-mail c.alonso@infor.uva.es
(***)   Dpt. Computing Science and Automatic Control, Faculty of Sciences,
          Open University UNED, c/ Senda del Rey, s/n, 08040, Madrid, Spain.
          TL . + 34 91 3987156, Fax + 34 91 3986697, e-mail fmorilla@human.uned.es
(****)   Azucarera Ebro Agrícolas (AEA), c/ Ruiz de Alarcón 5, Madrid, Spain
          TL. +34 91 5214650, Fax +34 91 5216407 manuel.bollain@aeasa.com

Abstract: An integrated advanced control and supervision system in operation in a sugar factory is presented. The system works on top of a commercial distributed computer control system, and combines artificial intelligence techniques for fault detection and diagnosis with advanced predictive controllers and models for other tasks.

Keywords: Hierarchical control, supervisory control, expert systems, fault detection and diagnosis, predictive control, process optimization.

1 Introduction

This paper gives an overview of a system for advanced and supervisory control in a beet sugar factory in Benavente, Spain.

The project uses artificial intelligence techniques and advanced control methods to improve the existing control system. In the factory, a distributed computer control system (Siemens   Teleperm) is in operation from several years on, providing what we can call the zero level in the hierarchy of control: standard regulatory control, alarms detection and an integrated operator's interface, giving a satisfactory performance in most of the cases.

Nevertheless, AEA, one of the biggest sugar companies in Spain, wanted to take advantage of the possibilities offered by modern technologies, and to test how them can help in the management of the production process. So the project was started in cooperation with other two partners: the University of Valladolid, and the Open University of Madrid.

Three main areas were a improvement was possible were identified:

- Fault detection and diagnosis of operation and equipment malfunctions
- Control of same complex processes
- Economical optimization of some operational set points

One method to try to automate the analysis of the plant in a given situation is to use expert systems. With this approach it is possible to incorporate in a computer the knowledge that the experts have on the process, and to treat it in a unified and consistent way. Even if the efficiency of these systems is below that of a human, there are several advantages that justify its use:

- They made a continuous supervision, 24 hours a day, with no lack of attention.
- The information from the process and the rules for interpreting it are treated in a uniform way, with no changes between shifts.
- They are able to deal with a big number of data in short periods of time.

Nevertheless, when a reliable mathematical model of the process is available, many decisions are better made by means of computations that use these models, taking into account all the interrelations among the variables of the process in a quantitative manner.

In practice, both kinds of knowledge of the process are always present: some is better express as rules and other can be synthesizing

more easily in mathematical models. So, the sensible approach when designing a supervisory system seems to be to try to combine both elements.

The system was organized in a hierarchical structure: in the lower level we maintain the existing distributed computer control (Teleperm) and its functions: direct digital control, displays, etc. On top, we implemented a supervisory layer that receives information of the plant from the Teleperm and it is able not only of providing advice about the recommended actions in a given situation, but also to act on the process by sending set points values to the controllers in the Teleperm as well as other orders. Physically that layer is placed in a workstation connected by a dedicated link to the Teleperm and also to a PC which collects data from the laboratory of the factory.

The supervisory functions are implemented in the workstation using an expert system shell, G2 from Gemsym Corporation, as well as a set of C programs for those tasks related to control and optimization problems. In Fig.1 we can see the hardware structure of the system.

In the remainder of the paper we describe the main functions performed by the system and its implementation in the factory. In section 2 we present the software architecture of the supervisory level and the supporting methodology. Sections 3, 4 and 5 are dedicated to its main modules: fault detection and diagnosis in the process, predictive control, and economical optimization. Finally, section 6 gives some conclusions extracted from the authors' experience.

2 System Architecture

The system, was conceived to respond to the above mentioned problems. It is built around and expert system (ES) that integrates its main functions: Supervision of process operation, fault detection and diagnosis, predictive control and optimization. The two firsts are implemented using the G2 shell, and the others as external programs that work under the control of the ES according to the state of the process.Fig.2 gives an overview of the system.
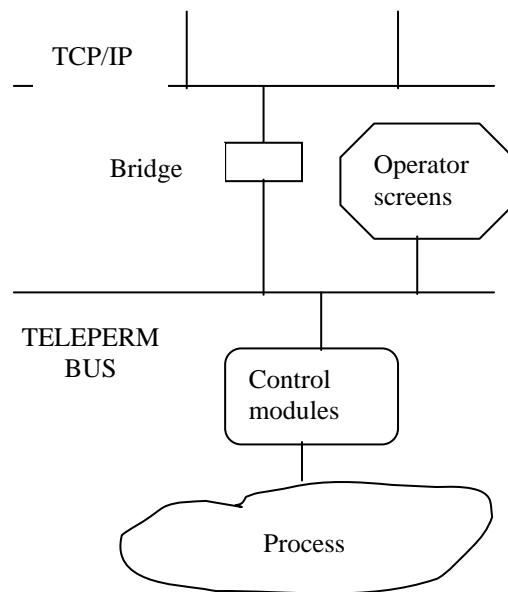

The knowledge and the operative behavior associated to every module are introduced using
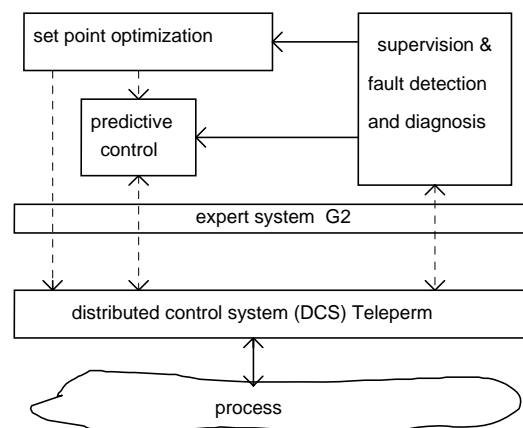

Fig.1 Hardware Architecture

Fig.2 Main blocks of the system

From the point of view of its internal architecture its is organized in a modular way, with a central working memory and a set of modules specialized in different functions that access the working memory when needed. This simplifies system development and maintenance, but, on the other side, it allows an extremely flexible control flow of information, that may make difficult to keep track of the system activity.

the standard tools of G2, such as frames and rules.

By default, process or laboratory data are requested by the modules of the system just when it need them. This request is not sent directly to the GSI module (external G2 interface), but to the working memory, or reasoning scheme, where there are objects with an associated frame containing the values and states of the variables of interest in its associated slots and a validity interval. If the working memory does not have the desired information, the consistency check module redirects the request to the GSI module. As soon as the data reach the system, the consistency check module validate them to avoid improper data to progress into the reasoning process, and then they are allocated into the working memory, from where the module that originated the request can read it.

Regarding modules activity, they can start up in several ways, depending on the module task nature. The interface module respond to operators' requirements and to some prespecified events, but, for instance, the monitoring module or the predictive control one have predefined scheduling times. When a function, like for instance Fault Detection, requires the participation of several modules, the coordination among them is event based.

It is worth to notice that the graphical schematics of the system user interface contains the interconnections among the constituent objects (tanks, pumps etc.) and that this information is used by the system to make almost the inferences about the process, resembling a blackboard architecture.

In relation with the implementation we tried to develop a prototype as soon as enough specifications have been stated, acting them as a kernel for a more sophisticated system. At the same time, we followed a methodology for the development, that can be summarize in the following way: Several sources of information, including meetings with the process experts, were used as the primary material to design the knowledge base and later on to implement the prototype. Once this was ready, and before to test it in the real process, we used a simulator to check the policies followed and to adjust many parameters that appear in the system. A similar methodology was followed with modules based on mathematical models.

For the purpose of testing, a dynamical non-linear mathematical model of the factory was developed. It combines first-principles with models obtained from experiments in the factory, and tries to reflect those aspects that are relevant for checking in real time the system, including fault detection and controller behavior. The model was simulated using the simulation language ACSL in a workstation connected under TCP/IP to another one where the ES was running, so that the simulation could play the same role as the set of distributed control system & process in the factory. In the workstation where the simulation runs it is possible to introduce changes in the process and to see how the system responds. ACSL did not work in real time, so a set of timing procedures, in addition to the communication software, were written for this purpose.

In relation with the test in the factory the policy was to develop the first prototype so that it include the main functions implemented at a limited scale, then to test each function in a particular section of the process and to introduce changes in the prototype according to the experience obtained. Finally to extend the reach of the improved system to the rest of the process.

3 Fault detection and Diagnosis

One important aim in the factory is to be able of detect and diagnose faults and problems in the process. Some of them are treated by the alarm system of the Teleperm, but other require more skill to detect and diagnose, and this is one of the reasons to introduce an expert system, taking into account that an early detection of a fault can help in avoiding more difficult situations in the future.

The system operates in the sections of diffusion, purification and evaporation in the factory, involving more that 400 variables under supervision.

These variables are called Monitored Variables (MV) inside the ES. A MV has three thresholds associated with it: trigger, confirmation and recovery; and three possible states: OK, vigilance and critical. As soon as it exceeds the first threshold, it becomes in vigilance state. Under this state, its sample rate is increased. If during some predefined time its value is over the confirmation threshold, it becomes in critical state. Once at this stage, a transitory object named "diagnosis" is created to find out the cause of the problem. Many diagnosis may be active, trying to confirm various causes as sources of trouble for MV.

To do it, we can take advantage of the symptoms observation to explain the presence of causes. Every monitored variable has associated a list of possible anomalies and every anomaly has associated a list of related effects that must evolve in time if they are really present. Each effect has assigned a temporal interval in which it must become observable. This is a kind of abductive reasoning, in which
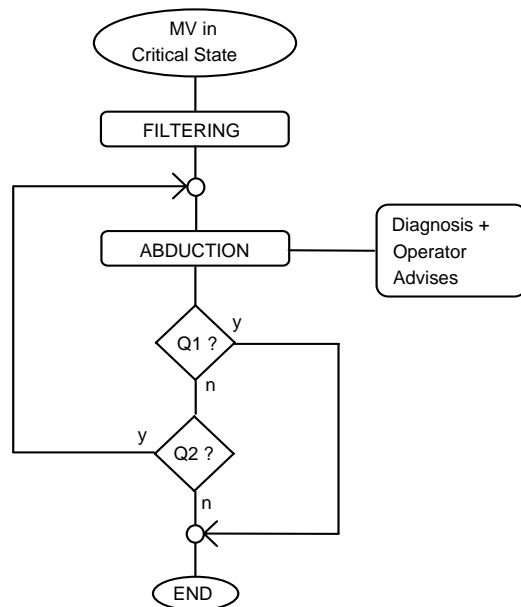
Q1: every observation related to every possible cause is done and there exists a diagnosis with a certainty greater than 0.7 (task condition).
Q2: there exists an observation that is still false but within the analysis interval (time condition).

Fig.3 Diagnosis algorithm

if a cause, C, has S1, S2 and S3 as effects, we assume that C is true if S1 ∧ S2 ∧ S3 is true. This is simple but it does not always works. A much more sound inference is obtained by rejecting the cause if any of the symptoms does not appear in a given time interval (filtering). The complete diagnostic strategy combines both approaches in a cyclic way: After the rejecting stage, the abductive method is invoked to sort the remaining candidate causes and also to rate the still not rejected causes in order to present the diagnostic results to the plant operator, arranged according to a possibilistic measure based on fuzzy logic. Time management was

introduced in the way shown by Fig.3 where we can see a scheme of the diagnosis algorithm. In the worse case, the cycle finishes when the maximum time of expiration of every effect is reached.

It is important to notice that the system is able to actuate directly over the process if, after giving the advice to the operator on how to drive the situation, he confirms the proposed solution.

7 Predictive Control

In order to improve the control of some parts of the process, we have introduced in our system a predictive control module, which operates in cooperation with the supervisory functions of the expert system.

Model Based Predictive Controllers (MBPC) use an internal dynamic model as a way to synthesize many aspects of the information we have about the process. The model is used to predict the future behavior of the process in response to manipulated variables (usually set points in the DCS). The future actions to be undertaken are computed using optimization procedures, in conjunction with the model in relation to the manipulated variables. The aim is to minimize a quadratic cost function of the errors between the predicted outputs of the plant and the required future evolution of them given as reference trajectories that connects the actual situation to the desired future setpoint. Its predictive nature allows implementing a system that can take actions to avoid unsafe future conditions or to obtain desired qualities via a reduced variance of the variables under consideration.

One important aspect to mention is that the computation can take into account the physical constrains of the process, using quadratic programming algorithms to solve the constrained optimization problem. Notice that, in this way, both dynamic of the plant and operating constrain can be handled within the same basic framework.

A receding horizon strategy is used to implement the control actions, so that the whole computation of a set of commands along the prediction interval is made continuously,
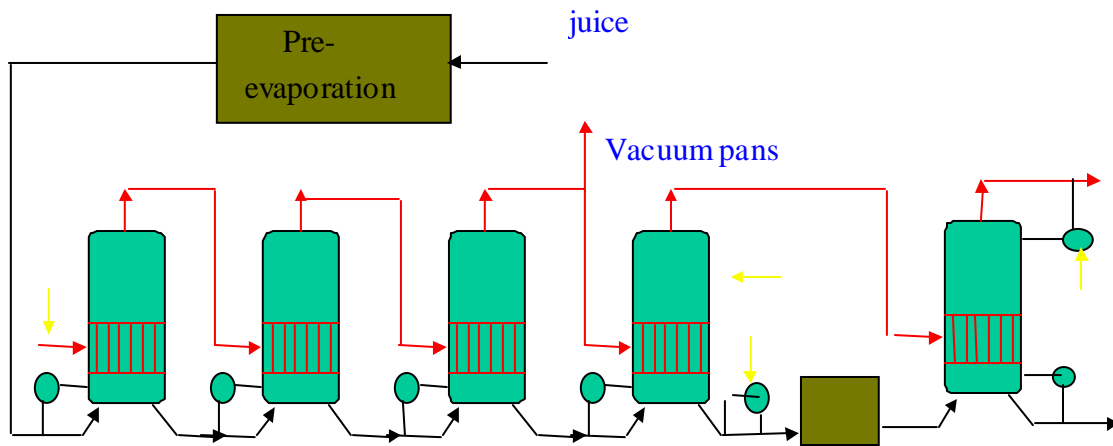
Fig.4 Evaporation station

but only the one corresponding to present time instant is applied.

The predictive controller is in operation in the evaporation section. This is a five effect one with 9 evaporators that can be seen schematically in Fig.4. This section concentrates by evaporation an incoming juice from a pre-evaporation tank, and provides steam for the vacuum pans of the factory. The main

Fig. 5 Time evolution of the main variables in
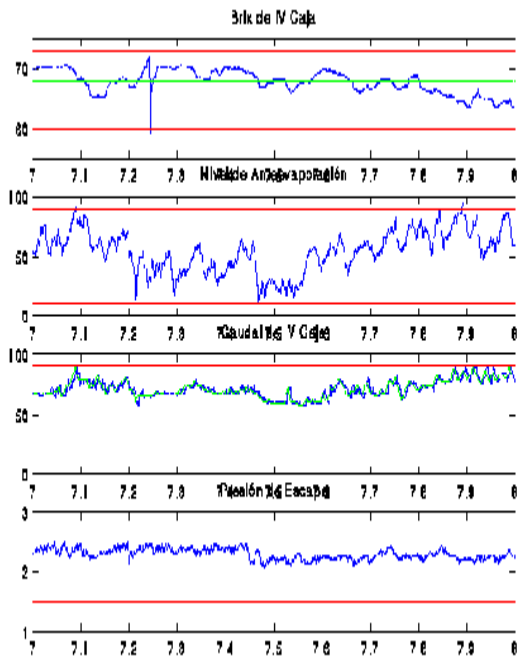
the evaporation

control aim is to maintain the concentration (brix) of the output juice and the level in the pre-evaporation tank, using as manipulated variables the output flow and steam pressure, which should move as less as possible, in spite

of disturbances due to vacuum pans demands and incoming flow and concentration. In practice changes in steam pressure are so limited that most of the time is not a practical MV.

The controller uses transfer functions as internal model that were identified using standard identification algorithms and a set of test plant. It is operating in a satisfactory way in the factory and helps in maintaining a constant product flow in the factory. In Fig. 5 one can see a record of the main variables for 10 hours: the two in the upper position are brix and level, that must be within the range (in red) specified by the operator. The other two are the manipulated variables output flow and steam pressure that must be kept as constant as possible. As we can see the controller is able to perform its duties quite well, taking into account the continuous disturbances that acts on the plant.

One important point to remark is the fact that the controller operates in conjunction with the expert system so that it can continue in operation under different situations. The main task of the expert system is to fix the conditions under which the controller must operate. This involves both faults in the process and different operating conditions in the factory.

The controller is able to adapt itself to different states of its variables, and to different aims. But these have to be provided by the expert system as part of its supervisory functions. As examples of the situations included in the rules of the ES we can mention: How to reconfigure the controller operation in the presence of faults in the process (lost of actuators, transmitters,

equipment malfunctions, ..)?. Which is the state of the slave controllers in the DCS?. What is the relevance of the conflicting different aims in a given situation?. Which is the future value of the disturbances?. What are the values to be applied to the constrains?. If there is no solution in the constrained optimization, which constrains must be removed?, etc.

8 Optimisation Functions

There are many set points to fix in the factory. Most of them are set according to technological or security reasons, but a few are directly related to the economics of the process and can be computed to optimise it for the operating conditions of the plant.

The approach we followed was to build an economical cost function which is the difference between the value of the sugar and pulp produced and the cost of fuel, electricity, coke, etc. This function can be maximised in relation to a few important degrees of freedom of the process: the amount of water introduced in the diffusion section, the operating steam pressure, and the recirculating syrup from the sugar end of the factory. The link between the variables and the cost function and the set points to be computed is given by the mathematical model of the plant. For this purpose we are using static mass and energy balances of the different sections. In these models appears a set of parameters, some of them like sugar extraction per beets, heat transfer coefficients, etc. are not known and change with time. To deal with these problem we organise this module so that it performs its tasks in a three steps sequence: Data from the DCS and the factory laboratory are first analyzed by the ES, to detect if, within a certain degree, they can be considered to be in steady state. If so, then average values in a period of time of 24 hours, are fed to an estimation algorithm which computes the values of the parameters of the static models so that their responses fit in the best way to the experimental data. It is worth to mention that in this stage a data reconciliation algorithm is also used. Finally, assuming that the parameters change slowly enough as to consider them constant for a period of time, the model can be used in the optimization procedure. This includes the actual operating constrains given by the ES. At present the only modules operating in the factory under test every eight hours, are the ones of the static model and data reconciliation. If the tests are satisfactory, then we will proceed ahead.

The optimum set points will be given to the DCS, operators or to predictive controllers for implementation as in Fig.2. and the whole procedure will be repeated when new conditions appear.

10 Conclusions

In this paper we have presented an overview of an integrated system for supervision and control of a sugar factory. The system is in operation with promising results. A key factor of success has been the cooperation between a team of people and partners with complementary backgrounds, the involvement of the staff of the sugar factory and the interest of the AEA directive team. It is not possible to mention all of them among the authors, but it is quite clear that the results obtained have been possible only with their contribution.

Bibliography

[1] "A simulator to validate fault detection in an industrial process with an expert system" L. F. Acebes, M.T. Alvarez, J. Achirica, C. Alonso, G. Acosta, C. Prada, Simulation of Continuous Systems, SCS-ESM'94, SCS, pp.709-713 Barcelona, 1994

[2] "Knowledge Based Diagnosis: Dealing with Fault Modes and Temporal Constrains".G. Acosta, C. Alonso, L.F. Acebes, A. Sanchez, C. Prada, International Conf. on Industrial Electronics, IECON'94, IEEE, pp.1419-1424, Bolonia, Italia, 1994

[3] " A Knowledge based approach to fault detection and diagnosis in industrial processes: a case study", C. Alonso, G. Acosta, C. Prada, J. Mira, International Symposium on industrial Electronics, ISIE'94, IEEE, pp.397-402 Santiago de Chile, 1994

[4] Advances in Model-Based Predictive Control. Edt. D. Clarke, Oxford Univ. Press, 1994

[5] HITO A tool for Hierarchical Control. C. Prada, S. Cristea, T. Alvarez, J.M. Zamarreño, T.J. Alvarez, 7th IEEE Emerging Technologies and Factory Automation , Barcelona, Octubre 1999