Fuzzy Supervision of Direct Controllers

Paulo Oliveira Pedro Lima João Sentieiro

CAPS/INIC and Instituto Superior Técnico (Technical University of Lisbon), Av.Rovisco Pais, 1, P-1096 Lisboa Codex, Portugal

Abstract

In closed loop control systems, a supervision loop is sometimes needed in order to perform an operation task which usually consists in adjustments of set-points and controller parameters.
Here a fuzzy approach to the supervision of controller parameters in single loop plants is described. The fuzzy supervision is performed over two different direct controllers: a PI and a Fuzzy Controller. Experiments were made to test the supervision strategy with two different simulated systems and one scaled pilot plant.
In the fuzzy controller, the central values of the rules output membership functions are adjusted at the supervision sampling instants. For the PI controller, the adjusted parameters are the proportional and integral gains.

Introduction

In complex processes, due to the inherent restrictions of control algorithms, the presence of human operators or automatic mechanisms of supervision is needed.
Supervision action can be approached in two senses. In a *strict* sense, related with controllers tunning, supervision is concerned with the change of controllers parameters in order to obtain a better closed loop performance.
In a wider sense, supervision is essentially concerned with monitoring of global process performance, related with process operation. Some interesting works have been presented [4], but a global approach is far from being established.
Knowledge based systems, together with pattern recognition, are useful tools to perform an automatic supervision procedure.
The *strict* approach has been addressed before by adaptive control, although that can not be considered a supervision method, because the change of controller parameters is not based on relevant features taken after some sampling instants but on the continuously sampled value of an error signal. However, some problems remain:

- Adaptive controllers have a considerable number of project parameters that are left free for the expert in control. So, the supervision of that kind of controllers is also an important field of investigation [1,2].
- In industrial process control, due to the classic solutions implemented and to the reluctance to change to complex control structures, PID controllers and other unsupervised direct controllers are still the most used.

In the following, a *strict* fuzzy supervision approach is taken and a supervision architecture for direct controllers, is proposed and discussed. A prior application of an algorithmic supervision architecture to fuzzy controllers has been presented by the authors in an earlier paper[8].
The framework to be used, with successful results in automatic control of complex industrial plants, such as cement kilns [6] and chemical processes[7], is the fuzzy control theory. Fuzzy inference systems [11] present advantages in what concerns to the knowledge representation. In order to validate the proposed supervision architecture, three systems were studied:

- A stable, linear minimum phase system, with an abrupt change in dynamics.
- A stable, linear non minimum phase system.
- A non-linear scaled pilot plant.

Two different controllers were used: a PI and a direct fuzzy controller. In those controllers, a supervisor was implanted and the improvement resulting from supervision is discussed.
Section 2 includes the description of the supervisor architecture and the implemented controller and supervisor loops. In section 3, results are presented, for both controllers with supervision. Finally some emergent conclusions are presented.

Proposed Architecture

Supervision Loop

Although classical direct controllers can achieve good performance in the control of linear systems, high non-linearities demand some kind of adaptiveness of the controller, namely if the working point of the process is time varying.
In the case of PI controllers the proportional and integral gains must be continuously monitorized and adjusted when system changes occur or when the working point changes.
In fuzzy controllers, which are non-linear in nature, the fine tuning of control rules is the key. Incorrect control actions are usually the result of ill-designed mathematical functions for the description of the linguistic terms used by the controller, or due to the existence of slightly incorrect premises for the control rules.

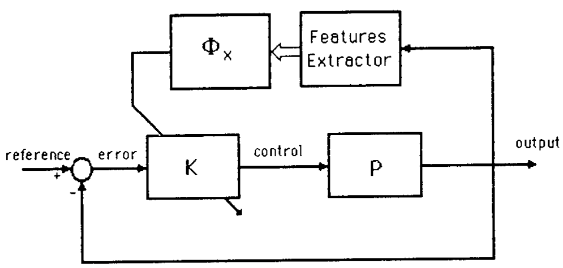


Figure 1: Architecture of the controller and supervisor

The overall architecture of the fuzzy supervised control loop is presented in figure 1. It is essentially a two loop hierarchical system where the basic loop is the control loop and the higher level loop is the supervision loop. The control loop with PI or fuzzy controller is similar to those found in the literature [5,7,10].
Typical fuzzy controller inputs are the *error* ($error(t) = reference(t) - output(t)$), the *change in error* ($\Delta error(t) = error(t) - error(t-1)$), the *integral of error* or other process features.
The supervisor function is represented in figure 1 by $\Phi_x(f_1, \ldots, f_n)$ where f_i, $i = 1, \ldots, n$ are the n features extracted from the process output. The block K is the basic loop controller.
The fuzzy supervisor is composed by three different elements:

- The *input fuzzy encoder* which consists of a set of analog membership functions, describing the *input linguistic terms.*
- The *linguistic control rules* , in the form **IF** *premises* THEN *actuation.* Here, the *premises* are described by the ***input linguistic terms*** (one for each input variable) and the ***actuation*** by the ***output linguistic terms.***
- A *defuzzifier*, which converts the output from the entire set of rules (determined by ***max-min*** fuzzy inference method) to a crisp control action.

From the point of view of the controller, given the system performance and a desired performance index, expressed by output features of the system, the supervisor performs a fine-tuning procedure.
In the experiences described later in the paper, the chosen features were:

- The overshoot
- The rise time percentage error (related to an estimated delay of the system output).

The two features are related to the system step response. In the fuzzy controller and due to the characteristics of the fuzzy inference procedure, a choice is done of only the most important rules fired. Based on the comparison between determined and desired features the supervisor will act upon those rules.

Controllers Implementation

Fuzzy Controller — The implemented fuzzy controller has two inputs: the error and the change in error between two consecutive instants.
Three linguistic terms are defined for each of the input variables: POSITIVE_BIG (PB), ZERO (ZE) and NEGATIVE_BIG (NB). Each linguistic term is described by the membership function $2^{-|x-a|}$, where a is the central value and x ranges in the universe of interest. Similarly, seven linguistic terms were defined for the output variable (the input of the controlled system): POSITIVE_BIG (PB), POSITIVE_MEDIUM (PM), POSITIVE_SMALL (PS), ZERO (ZE), NEGATIVE_SMALL (NS), NEGATIVE_MEDIUM (NM) and NEGATIVE_BIG (NB).

Table 1: Rules protocol for fuzzy controller

		Change in error		
	δu	PB	ZE	NB
Error	PB	PB	PM	NS
	ZB	PM	ZE	NM
	NB	PS	NM	NB

The initial control protocol, a set of nine rules, can be seen in table 1.
This protocol agrees with some constraints related with symmetry and stability [3,8]. The outputs of the table are the seven linguistic terms associated with the control variable.

The defuzzification is done by the simplified centroid formula:

$$y = \frac{\sum_{xx} \overline{V}_{xx}\, \mu_{xx}(\overline{V}_{xx})}{\sum_{xx} \mu_{xx}(\overline{V}_{xx})} \qquad (1)$$

where the values of $\overline{V}_{xx}$ are the central values of the output membership function μ_{xx} associated with the output linguistic term xx, after scaling by the *min-max* fuzzy inference method.

PI Controller — The PI controller is implemented by

$$u(t) = K_p(error(t) + K_i \sum_{k=t_0}^{t} error(k)) \qquad (2)$$

In the experiments, K_p and K_i are initially mistuned, in what concerns to the desired output features.

Fuzzy Supervisor of Fuzzy Controller

The features chosen as supervisor inputs were *overshoot* and *rise time*, as described before.

Table 2: Rules protocol for fuzzy supervisor of fuzzy controller

		t_r		
	$\delta\overline{V}_{xx}$	PB	PM	PS
$S\%$	PB	PSs	NMs	NBs
	PM	PMs	ZEs	NMs
	PS	PBs	PMs	NSs

The operation of the fuzzy supervisor is based on the protocol presented in table 2. From the table, the change in $\overline{V}_{xx}$, $\delta\overline{V}_{xx}$ is determined. New values for central values of output membership functions are then computed as

$$\overline{V}_{xx}(k+1) = \overline{V}_{xx}(k)(1 + \delta\overline{V}_{xx})$$

The set of most important rules during the last considered period is required, because only those rules are changed.

The defuzzification procedure used in the supervisor is also the centroid method.

Different options were considered in order to choose the supervisor actuation instants:

- Constant supervision sampling time (greater than system sampling time).
- Variable supervision sampling time, related to the reference input changes.
- Variable supervision sampling time, given by a delay after steady-state of the step response.

Assuming the system to be controlled as stable, the last option has been chosen, since it is the most closely related with the instants when new information about feature values is known.

The displacement decisions fulfill the restrictions presented in [3] and [8], which intend to preserve the initial policy of linguistic control used by the operator.

Fuzzy Supervisor of PI Controller

The features chosen as supervisor inputs are the same as for the fuzzy controller.

Table 3: Rules protocol for supervision of K_i

		t_r		
	δK_i	PB	PM	PS
$S\%$	PB	NBi	NMi	NSi
	PM	ZEi	ZEi	ZEi
	PS	PSi	PMi	PBi

Table 4: Rules protocol for supervision of K_p

		t_r		
	δK_p	PB	PM	PS
$S\%$	PB	PBp	ZEp	NSp
	PM	PMp	ZEp	NMp
	PS	PSp	ZEp	NBp

The protocol rules which perform the changes in K_i and K_p parameters resulted from simulations performed with second order systems closed loop controlled by a PI and are presented in tables 3 and 4.

The defuzzification procedure used in the PI supervisor is also the centroid method.

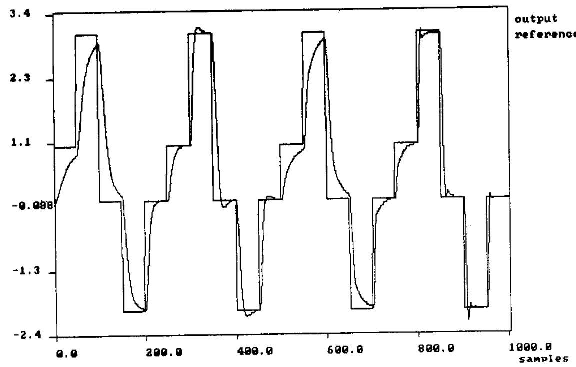
The option of variable supervisor sampling time has been chosen, as in the previous case.

Results

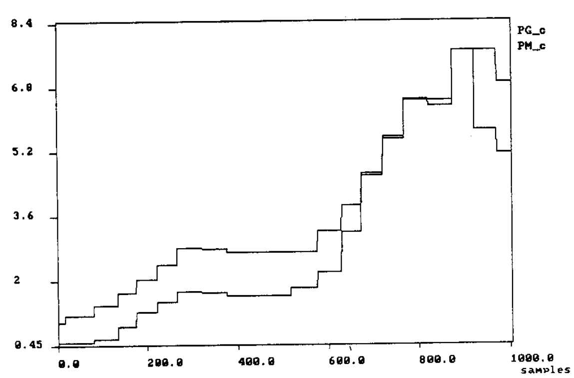
The validation of the supervision method has been done by testing the results of its application over two simulated systems and a scaled pilot plant. For each of them, the supervised PI and the supervised fuzzy controller were compared.

The objectives of the study were:

- Testing the robustness of the fuzzy supervision with respect to changes in dynamics, non-linearities or "difficult" systems;
- Evaluating convergence speed and steady-state behavior of the systems responses to a sequence of steps, denoted by the evolution of features and controller parameters.



a) supervised fuzzy controller



b) evolution of central values of the most important output membership functions

Figure 2: Results for the minimum phase system with change in dynamics and fuzzy controller

In the simulations, undisturbed stable systems were assumed.

The first simulation was performed over a linear 2nd. order discrete-time system, with minimum-phase and a change in dynamics at the middle of the simulation.

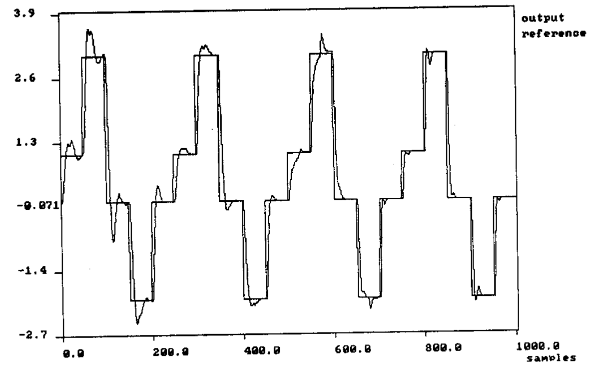
The initial difference equation describing the system was

$$\begin{aligned} y(t) &= 1.06y(t-1) - 0.22y(t-2) + \\ &\quad 1.99E - 2u(t-1) + 1.99E - 2u(t-2) \end{aligned}$$

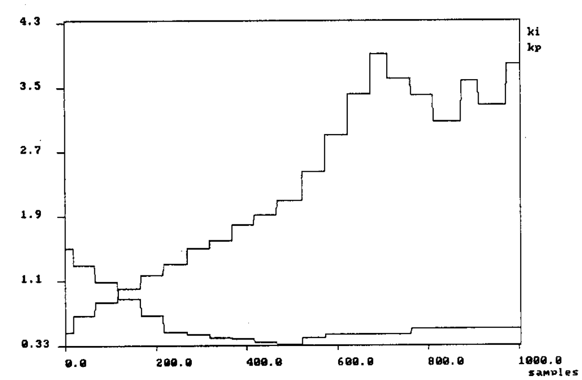
and after change in dynamics

$$\begin{aligned} y(t) &= 0.9y(t-1) - 0.22y(t-2) + \\ &\quad 1.99E - 2u(t-1) + 1.99E - 2u(t-2) \end{aligned}$$

The results are shown in figures 2 and 3. It can be seen that both controllers are robust to the system change in



a) supervised PI controller



b) evolution of K_i and K_p

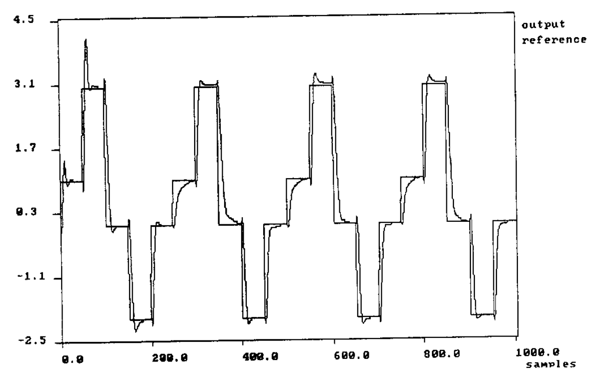
Figure 3: Results for the minimum phase system with change in dynamics and PI controller

dynamics, due to the fuzzy supervision. There is a slight oscillation in the steady state values of the PI controller gains which results from the trade-off between the target rise-time and overshoot values. Notice that for a required 0% overshoot the value of rise-time is lower-bounded. However, changes in controller parameters result in contradictory evolutions for the rise-time and overshoot of step response. The more overshoot is diminished , the more rise-time grows.

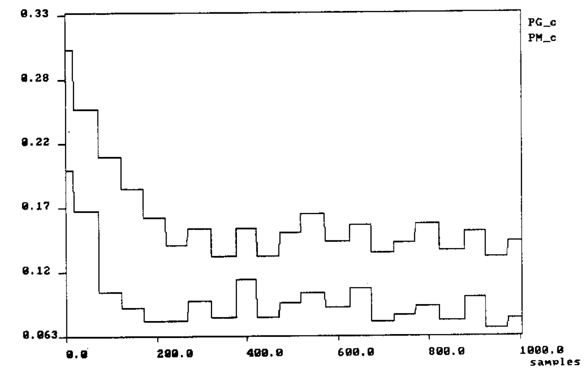
The other simulated system was also linear and 2nd. order discrete-time, but now with a zero outside the unit circle, that is, a non-minimum phase system, described by

$$\begin{aligned} y(t) &= 1.2y(t-1) - 0.35y(t-2) - \\ &\quad u(t-1) + 2u(t-2) \end{aligned}$$

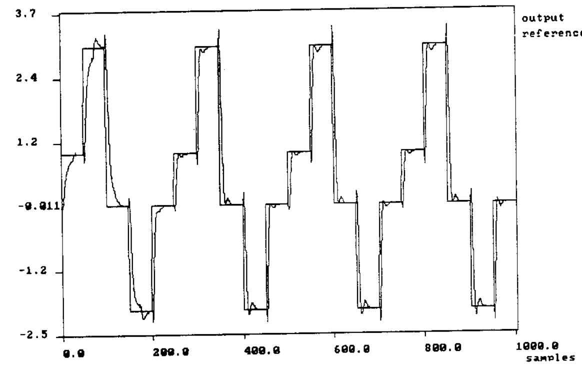
Non-minimum phase systems are difficult to control



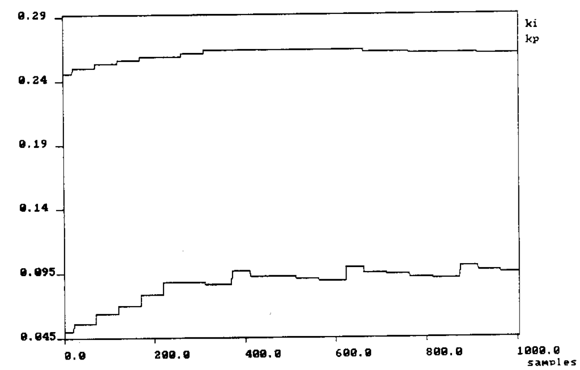
a) supervised fuzzy controller



b) evolution of central values of the most important output membership functions



a) supervised PI controller



b) evolution of K_i and K_p

Figure 4: Results for the non-minimum phase system and fuzzy controller

Figure 5: Results for the non-minimum phase system and PI controller

due to its characteristic step-response: first the output evolves in the opposite direction of the applied step and only after a few instants follows that direction.

From figures 4 and 5 it can be noticed that, although oscillations are presented by controller parameters in the fuzzy controller case, the supervisor deals with the tuning task, necessary to achieve the desired features.

The non-minimum phase characteristic of the controlled system explains the oscillations, namely for the fuzzy controller. This is due to the lack of precision of the fuzzy controller in dealing with the errors presented by this type of system, which cover a broader range of values in the universe of interest for the membership functions involved. An alternative could be the use of more input linguistic terms and consequently a more detailed set of rules.

Finally, the strategy has been tested with a scaled pilot plant. This was a tank system with sump and process tanks, circulating pump, variable area flowmeter and motorized flow control valve. An additional manual flow control valve allows the process tank draining adjustment. There is also a level sensor which measures the liquid level inside the tank, drawn from the sump tank by a centrifugal pump, at a rate controlled by the motorized valve and visually measureable by the flowmeter.

The purpose of the experiments was to control the liquid level inside the tank. The resultant system is non-linear, namely because of the characteristic flow/input current in the motorized valve, including hysteresis.

Once again, the results presented in figures 6 and 7 show the convergence of the supervision procedure to the desired features after a few step responses.

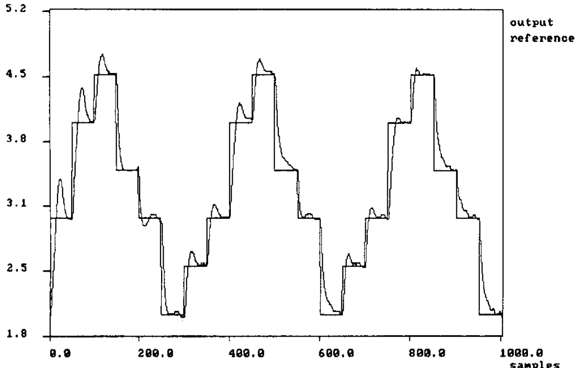


Figure 6: Results for the scaled pilot plant with supervised fuzzy controller

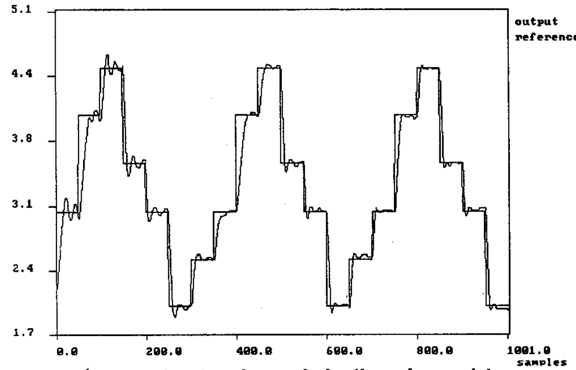


Figure 7: Results for the scaled pilot plant with supervised PI controller

Conclusions and Future Trends

A new strategy for supervision of fuzzy and PI controllers was presented. It consists of a continuous adjustment of controller parameters. The amount of adjustment results from a fuzzy rules based inference which take into account two features extracted from the control system output - the rise time and the overshoot.
In the case of fuzzy controllers this strategy differs from other known approaches to fuzzy supervision [3,9] in the sense that the linguistic meaning of the rules and its relation to initial operator actions are preserved.
It is also shown that PI controllers can be tuned on line automatically by heuristic rules, based on features of the system output.
In both cases, experimental results show that the strategy is robust relatively to changes in dynamics, nonlinearities and "difficult" systems such as non-minimum-phase. No proof of convergence of the method has been presented, but it has been shown in all the examples that the target features were achieved after a few supervision sampling instants.
At the present in the fuzzy controller, every intervenient rule classified among the most important ones is adjusted by the same amount. In the future, displacements weighted by the relative importance of the rules will be considered.

References

[1] Årzén, K.-E. (1989). *An Architecture for Expert System Based Feedback Control.* Automatica, vol. 25, N$^{\underline{o}}$ 6, pp. 813-817.

[2] Åström, K. J.; Anton, J. J. e Årzén, K.-E. (1986). *Expert Control.* Automatica, vol. 22, N$^{\underline{o}}$ 3, pp. 277-286.

[3] Cerezo, A. J. G. (1987). *Aplicaciones del Razonamiento Aproximado en el Control y Supervision de Processos.* Ph. D. Thesis, ETSII de Vigo, Febrero.

[4] Doraiswami, R. and Jiang, J. (1989). *"Performance Monitoring in Expert Control Systems"* Automatica, vol. 25, N$^{\underline{o}}$ 6, pp. 799-811.

[5] Kickert, W. J. M. e Van Nauta Lemke, H. R. (1976). *Application of a Fuzzy Controller in a Warm Water Plant.* Automatica, vol. 12, pp. 301-308.

[6] King, R. E. e Karonis, F. C. (1988). *Multi-Level Expert Control of a Large-Scale Industrial Process.* Fuzzy Computing, pp. 323-339.

[7] King P. J., Mandani E. H. (1977). "The Application of Fuzzy Control Systems to Industrial Processes". Automatica, vol. 13, pp. 235-242.

[8] Oliveira, P.; Lima, P.; Sentieiro, J. J.; Sanz, R.; Galan, R. e Jimenez, A. (1990). *An Architecture for the Supervision of Fuzzy Controllers.* Proceedings of IEEE International Workshop on Intelligent Robots and Systems, IROS'90, July 1990.

[9] Procyk, J. J. e Mandani, E. H. (1979). *A Linguistic Self-Organizing Process Controller.* Automatica, vol. 15, pp. 15-30.

[10] Sugeno, M. (1985). *An Introductory Survey of Fuzzy Control.* Information Sciences N$^{\underline{o}}$ 36, pp. 59-83.

[11] Zadeh, L. A. (1973). *Outline of a New Approach to the Analysis of Complex Systems and Decision Processes.* IEEE Trans. Systems, Man and Cybernetics, vol. SMC-3, N$^{\underline{o}}$ 1, January.