Fuzzy Supervision on Intelligent Control Systems

Paulo Oliveira Pedro Lima João Sentieiro

CAPS/INIC and Instituto Superior Técnico (Technical University of Lisbon)

Av.Rovisco Pais, 1

1096 Lisboa Codex, Portugal

Abstract

In intelligent control systems, an additional loop of supervision is sometimes needed to perform an adaptation which usually consists on set-points adjustments and controller parameters tuning.

This paper describes a fuzzy approach to the supervision of controller parameters, in single loop plants. The fuzzy supervision is performed over a PI controller. In order to evaluate this supervision strategy two different simulated systems and one scaled pilot plant were tested.

Keywords – Intelligent Control, Supervision, Fuzzy Logic, Fuzzy Control, Fuzzy Supervision.

1 Intelligent Control Systems

Due to the inherent restrictions of control algorithms the presence of human operators or automatic mechanisms of supervision is of major importance when controlling complex processes [Årzén89, Åström86].

During the last decade, research has been done in the area of autonomous systems. Faced with the new problems arising in the area, the control community has been discussing and proposing new approaches.

Adaptive control was an early attempt to increase classical controllers autonomy. However, the structure of the controller has to be chosen a priori and its parameters must be selected. Therefore, specialized control design methods have to be used in order to implement adaptive controllers.

Furthermore problems like world perception, decision strategies and planning are not covered by adaptive control or other control systems design methodologies.

An important conclusion, in what concerns the design of an autonomous system, is the generally accepted fact that some intelligence has to be incorporated in the overall control system, in order to insure the system survival in an unknown/aggressive environment.

The establishment of a unified theoretical framework, where these problems can be formalized and solved, has been the major goal of a new research area known as Intelligent Control [Meystel88, Saridis83].

A.I. has developed methodologies that can be used on the synthesis of basic "intelligence" and that provide means of representing the world (based on KBS, semantic networks, frames), of planning (based on heuristic searches as depth first, A*), of infering new knowledge about the mission to be performed and of executing and/or changing the generated plans.

In this paper an architecture for the supervision of controllers supported on the Fuzzy Logic Theory, is proposed.

Fuzzy logic has been successfully used, in the automatic control of complex industrial plants, such as cement kilns [King88] and chemical processes [Mandani77], analysis of Driver-System environment [Kramer85] and on aircraft control [Chiu90]. Fuzzy inference systems [Zadeh73] offer advantages relatively to the conventional systems, in what concerns the incorporation of *possibility* information, obtained from the world by sensing, and *possibilistic* inference [Valavanis90], from human operators or self-organizing mechanisms.

In section 2, an introduction to fuzzy logic and linguistic fuzzy control basic concepts is presented. In Section 3 a fuzzy supervisor for a PI controller is proposed. In Section 4 the performance of the proposed supervision architecture is tested for three relevant types of systems. Finally, Section 5 concludes the paper pointing out directions for further research.

2 Concepts of Linguistic Fuzzy control

A fuzzy subset A of an universe of discourse (support set) U is characterized by a membership function $\mu_A(x) : x \in U \rightarrow [0, 1]$, representing the grade of membership of x in A [Zadeh73].

Each word or linguist term in a natural language can be viewed as a label for a fuzzy subset A of a universe of discourse U. This language assigns atomic and composite labels describing words, phrases and sentences to subsets of U [Zadeh73].

A fuzzy linguistic variable is a variable whose values are linguistic terms used as labels of fuzzy sets. For instance, the *atomic* fuzzy subset labels *high*, *medium*, *low* and *ok* can be regarded as values of the fuzzy variable *temperature*.

The three basic operations among fuzzy sets (complement, union and intersection) are described in terms of the membership functions for the intervening sets. They are related to the negation *not* of labels and to connectives *or* and *and*, between labels.

- **complement**: $\mu_{\neg A}(x) = 1 - \mu_A(x)$
- **union**: $\mu_{A\cup B}(x) = max[\mu_A(x), \mu_B(x)]$
- **intersection**: $\mu_{A\cap B}(x) = min[\mu_A(x), \mu_B(x)]$

A controller can be interpreted as a system that maps an input signal to an output signal. A fuzzy conditional statement describes this mapping, by a set of production rules, based on the knowledge representation.

For a controller with two inputs and one output, a typical fuzzy conditional statement (or fuzzy rule) is:

IF V_1 is T_1 and V_2 is T_2 **THEN** V_o is T_o

where

V_i ,$i = 1, 2$, is the linguistic variable for the input i;

T_i ,$i = 1, 2$, is one of the linguistic terms assumed by V_i;

V_o,T_o have the same meaning for the controller's output.

Given the fuzzy subsets $A \subset U$ and $B \subset V$, a fuzzy conditional statement R, from U to V, of the form:

IF A THEN B

is defined by the bivariate membership function

$$\mu_R(x,y) = min[\mu_A(x), \mu_B(y)], \quad x \in U, y \in V \qquad (1)$$

Given the relation R, from U to V, and A' a fuzzy subset of U, the fuzzy subset B' of V inferred from A', has the membership function

$$\mu_{B'}(y) = \max_x \ min[\mu_{A'}(x), \mu_R(x,y)] \qquad (2)$$

that results of the application of the **compositional rule of inference**(CRI) [Zadeh73].

In this work, the controller input values are crisp rather than fuzzy sets. Therefore, the CRI can be simplified by interpreting any input x_0 as a fuzzy input set A' with the membership function

$$\mu_{A'}(x) = \begin{cases} 0 & x \neq x_0 \\ 1 & x = x_0 \end{cases}$$

that yields

$$\mu_{B'_r}(y) = min[\mu_A(x_0), \mu_B(y)] = \mu_R(x_0, y) \qquad (3)$$

for a particular fuzzy rule r [Kickert76].

The fuzzy controller is defined by a set of rules with the form just described. The final output fuzzy set, resulting from an input x_0 and an inference cycle over all the rules, is given by

$$\mu_{B'}(y) = \max_r \ min[\mu_A(x_0), \mu_B(y)] \qquad (4)$$

Similarly as considered for input values, the output must be a crisp value. The **centroid method** has been chosen for this *defuzzification* operation:

$$u = \frac{\int \mu_{B'}(y) y dy}{\int \mu_{B'}(y) dy} \qquad (5)$$

3 Proposed Architecture

Although classical direct controllers can achieve good performance in the control of linear systems, in the case of PI controllers the proportional and integral gains must be continuously monitored and adjusted when system dynamics changes occur or when the working point varies. This means that when PI controllers are implemented some adaptation mechanisms need to be incorporated in the Supervision level.

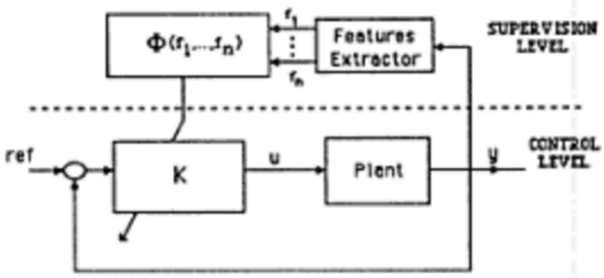


Figure 1: Architecture of the controller and supervisor

In the supervision architecture proposed in [Oli90], the supervisor was implemented through an algorithm. In the present work a fuzzy approach is used at the supervision level. The overall architecture of the fuzzy supervised control loop is presented in figure 1.

Typical fuzzy supervisor inputs are the *rise time*, the *settling time*, the *overshoot*, the *output offset*, the *activity in the control* or other process features. The supervisor function is represented in fig. 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. A good reason to use features instead of a model reference scheme, is the fact that they provide information which is closer to the one used by an human operator and does not impose a mathematical model to the output response. Even though, there is a trade-off between performance and complexity of the controlled system. In systems of higher complexity, this approach can be seen as a step towards the use of *perception*. In this context, the sensor numerical values are replaced by symbolic data and fuzzy rules can be understood as a simple planner.

In what follows the *overshoot* ($S\%$) and the *rise time* (t_r) are the chosen features.

Table 1: 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

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* of the input features – overshoot and rise-time – related with the system step response.

- 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 feature) 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 supervision actuation and that is implemented by the simplified version of (5):

$$u = \frac{\sum_i \mu_{B'}(y_i) y_i}{\sum_i \mu_{B'}(y_i)}$$

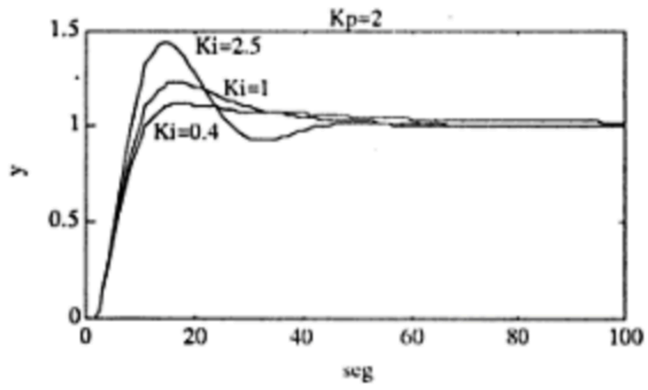


Figure 2: Simulation results with constant K_p

The protocol rules, which perform the changes in the PI controller parameters K_i and K_p are presented in tables 1 and 2 and they were established with basis on simulations performed on second order systems (figs. 2 and 3)

In order to choose the supervisor actuation instants different options were considered:

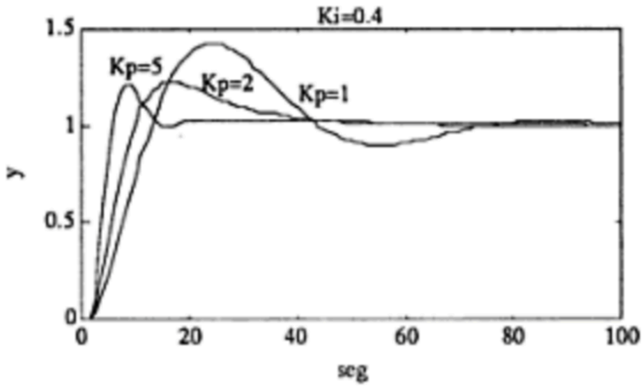


Figure 3: Simulation results with constant K_i

Table 2: 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

- Constant supervision sampling time (greater than system sampling time).

- Variable supervision sampling time, related to the reference input changes.

- Variable supervision sampling time, defined by a fixed delay with respect to the time instant when the step response reachs a 5% deviation of the steady-state.

The last option was implemented, together with a time-out mechanism for the tuning of systems with very slow output response.

From the point of view of the controller, given the actual system performance and a desired performance index expressed by output features of the system, the supervisor performs a fine-tuning procedure.

4 Results

To validate the proposed supervision architecture, three systems were studied:

- A stable, linear and minimum phase system with an abrupt change in its dynamics.

- A stable, linear and non minimum phase system.

- A non-linear scaled pilot plant.

The PI controller is implemented by

$$u(t) = K_p(error(t) + K_i \sum_{k=t_0}^{t} error(k)) \qquad (6)$$

In the experiments, K_p and K_i are initially mistuned and undisturbed stable systems were assumed.

The objectives of the study were:

- to test the robustness of the fuzzy supervision with respect to changes in dynamics, non-linearities or non-minimum phase systems;
- to evaluate the convergence rate and the steady-state behaviour of the systems responses to a sequence of steps, exhibited by the evolution of the features and the controller parameters.

The first test was run for a simulated linear second order discrete-time system, with minimum-phase and experiencing a change in its dynamics at the middle of the simulation time interval.

The difference equations describing the system before and after dynamics change are respectively:

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

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

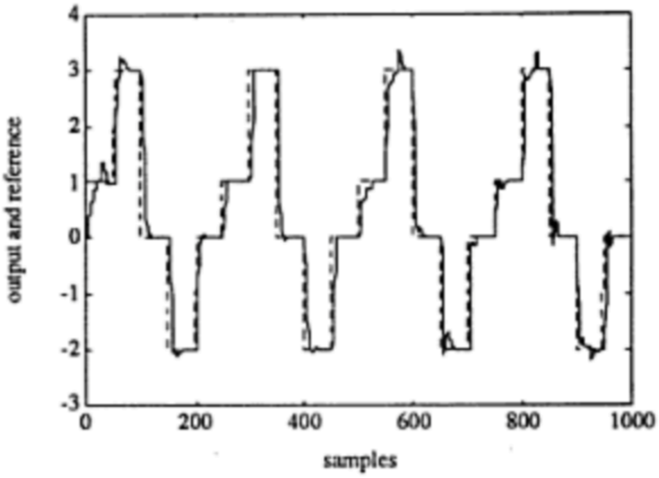
The results are shown in figure 4. It can be seen that the controller is robust to the change in dynamics, due to the fuzzy supervision. There is a slight oscillation in the steady state values of the PI controller gains which is due to the trade-off between the target rise-time and the overshoot values. Notice that for a required 0% overshoot the value of the rise-time is lower-bounded. However, changes in controller parameters result in contradictory evolutions for the rise-time and for the overshoot to the step response. To a decrease on the overshoot corresponds an increase on the rise-time.

The other simulated discrete-time system was also linear and second order, but exhibiting one zero outside the unit circle, that is, a non-minimum phase system, described by

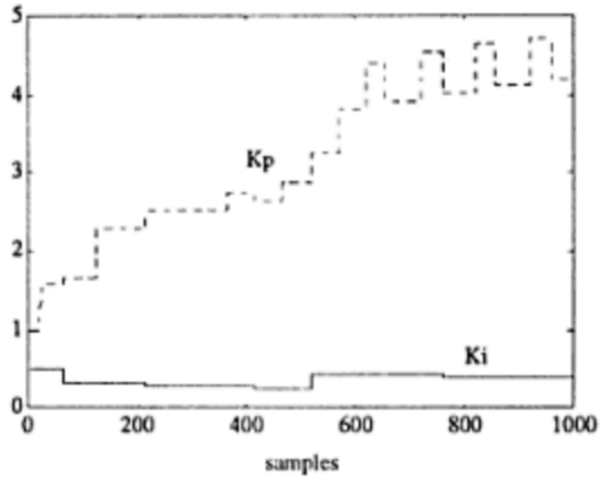
$$\begin{aligned} y(t) &= 1.2y(t-1) - 0.35y(t-2) - \\ &\quad u(t-1) + 2u(t-2) \end{aligned}$$

In figure 5 it can be noticed that in this case the supervisor improves the controller performance during adaptation.

Finally, the strategy has been tested with a scaled pilot plant consisting of a tank system with sump and process tanks, circulating pump, variable area flow meter and a motorized flow control valve. An additional



a) supervised PI controller



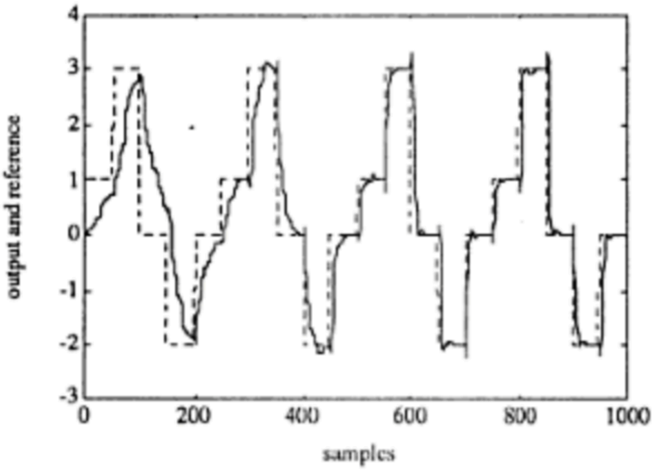
b) evolution of K_i and K_p

Figure 4: Results of using a PI controller on a minimum phase system with change in dynamics

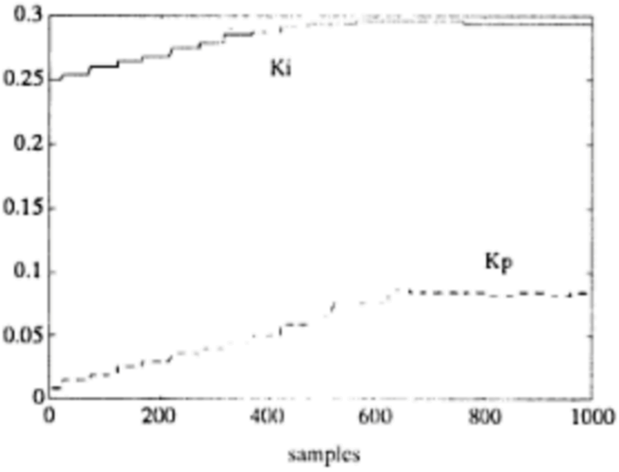
manual flow control valve allows the draining adjustment of the process tank. There is also a level sensor which measures the liquid level inside the tank, drawn from the sump tank by a centrifugal pump, at a rate controlled by the motorized valve and visually measurable by the flowmeter.

The purpose of the experiments was to control the liquid level inside the tank. The resultant system is non-linear, namely because of the flow/input current characteristic of the motorized valve (that includes among others an hysteresis non-linearity).

Once again, the results presented in figure 6 show that the supervision procedure converges to the desired features after a few step responses.

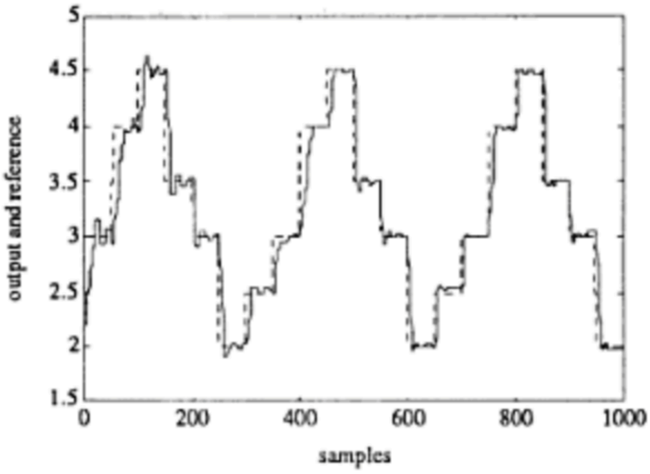


a) supervised PI controller

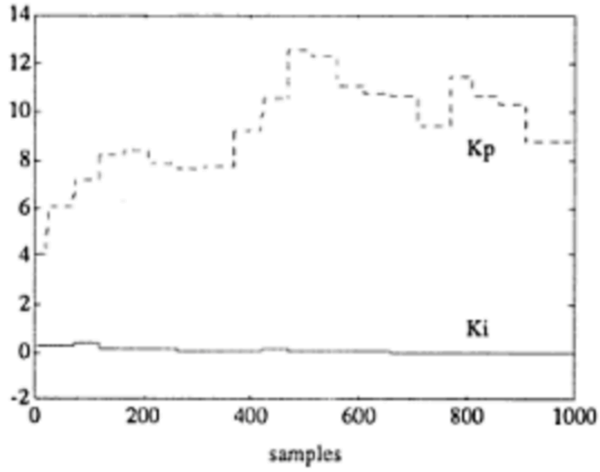


b) evolution of K_i and K_p

Figure 5: Results of using a PI controller on a non-minimum phase system



a) supervised PI controller



b) evolution of K_i and K_p

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

5 Conclusions and Future Trends

A new strategy for supervision of PI controllers was presented. It consists of an adjustment of PI controller gains. The amount of adjustment results from a fuzzy rule based inference which takes into account features extracted from the controlled system output - the rise time and the overshoot.

It is also shown that PI controllers can be tuned online, automatically, by heuristic rules based on features of the system output.

The experimental results reveal that the strategy is robust relatively to changes in dynamics, non-linearities and non-minimum phase systems. No proof of convergence of the method has been presented, but it has been shown for all the examples that the target features were achieved after a few supervision sampling instants.

In the future, effort will be focused on proving convergence of the proposed supervision architecture.

References

[Årzén89] Årzén, K.-E. (1989). "An Architecture for Expert System Based Feedback Control". Automatica, vol 25, n 6, pp 813-817.

[Åström86] Åström, K. J.; Anton, J. J. e Årzén, K.-E. (1986). "Expert Control". Automatica, vol 22, n 3, pp 277-286.

[Cerezo87] Cerezo, A. J. G. (1987). "Aplicaciones del Razonamiento Aproximado en el Control y Supervision de Processos". Ph. D. Thesis, ETSII de Vigo, Febrero.

[Chiu90] Chiu S., Chand S., Moore D. and Chaudhary A. (1990). "Fuzzy Logic-Based Flexible Wing Aircraft Roll and Structural Torsion Moment Control". Proceedings of the 5^{th} IEEE International Symposium on Intelligent Control". pp 897-902.

[Kickert76] 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.

[King88] King, R. E. e Karonis, F. C. (1988). "Multi-Level Expert Control of a Large-Scale Industrial Process". Fuzzy Computing, pp 323-339.

[Kramer85] Kramer, U. (1985). "On the Application of Fuzzy Sets to the Analysis of the System Driver Vehicle Environment". Automatica, vol 21, n 1, pp 101-107.

[Mandani77] King P. J., Mandani E. H. (1977). "The Application of Fuzzy Control Systems to Industrial Processes". Automatica, vol 13, pp 235-242.

[Meystel88] Meystel, A. (1988). "Intelligent control in robotics". J. Robotic Systems, vol 5, pp 269-308.

[Oli90] 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.

[Procyk79] Procyk, J. J. e Mandani, E. H. (1979). "A Linguistic Self-Organizing Process Controller". Automatica, vol 15, pp 15-30.

[Saridis83] Saridis, G. N. (1983). "Intelligent Robot Control". IEEE Trans. on AC, vol. AC-28, n 5.

[Sugeno85] Sugeno, M. (1985). "An Introductory Survey of Fuzzy Control". Information Sciences n 36, pp 59-83.

[Valavanis90] Valavanis, K. and Saridis, G. (1990). "A Review of Intelligence Control Based Methodologies for Modeling and Analysis of Hierarchically Intelligent Systems". Proceedings of the V International Sysmposium on Intelligent Control, pp 15-20.

[Zadeh73] 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 3, n1, January.