An Improved Self-Tuning Mechanism of Fuzzy Control by Gradient Descent Method

Ahcène Habbi*, Mimoun Zelmat
Laboratoire d'Automatique Appliquée, University of Boumerdès, FHC
35000 Boumerdès, Algeria.
*Corresponding author. Tel./Fax: +213-24-816905, E-mail: habbi_hacene@hotmail.com

Abstract – An improved self-tuning mechanism of fuzzy control by gradient descent method is presented. The membership function parameters are tuned by minimizing some criterion defined on the control output using the steepest gradient descent algorithm. The factor which controls how much the fuzzy controller parameters are altered is adjusted continuously using a set of fuzzy rules. This varying factor is determined with respect to the values of the objective function and its change. An application to the control output optimization of a PI-type fuzzy controller shows the superiority of the proposed self-tuning mechanism over a previously published approach in terms of both precision and convergence rate.

Keywords: adaptive fuzzy control; fuzzy reasoning; gradient descent algorithm; self-tuning.

I. INTRODUCTION

Recently, the "fuzzy logic wave" has reached the community of automatic control. Fuzzy logic controllers (FLC's) have been successfully used for a number of complex, ill-structured industrial processes [KIR, 98; FIS, 99]. Since most of the real-world processes that require automatic control are nonlinear in nature, FLC's can be designed to cope with a certain amount of process nonlinearity and parameter variations. Therefore, more attention has been paid to the problem of how to design a suitable adaptive fuzzy controller for a given process. Different types of adaptive FLC's have been developed and implemented for various practical applications [DRI, 96]. Adaptation mechanisms for FLC's are classified according to which the controller parameters are adjusted. Adaptive controllers that adjust the fuzzy set definitions or scaling factors are called self-tuning controllers (STC). However, when the fuzzy rule base is altered, the controller is called self-organizing controller (SOC).

Many works have centered on the use of mathematical optimization techniques (see, [BOR 90]) to tune the set definitions so that the output from the FLC matches a suitable set of reference data as closely as possible [DRI, 96; WON, 98; HE, 93]. The basic example of this is given by Nomura et al.. in [NOM, 91], where they use gradient descent algorithm to tune simple membership functions. The controller is tuned iteratively by minimizing the square error between the FLC output and the desired output given by the training data. This tuning method may be very good for control systems, but its applicability is closely related to the convergence rate of the adaptation algorithm, especially when it is used on-line as Glorennec did in the control of a mixer tap [GLO, 91]. The main problem is how to adequately choose the constant which controls how much the controller parameters are altered at each iteration in the gradient descent algorithm. This, however, puts an unnecessary and often inappropriate constraint on the design. A suitable choice of the gradient step may accelerate the convergence of the algorithm (see, for example [SAD, 75]) and then enhance the performance of the fuzzy control loop.

This limitation of the self-tuning mechanism of fuzzy control suggested by Nomura et al. motivated us to investigate techniques of improving the original algorithm by using experts' knowledge rather than mathematical models or heuristics methods. In the modified version the gradient step is adjusted continuously with the help of IF-THEN fuzzy rules. The amount of variation of this factor is determined with respect to the current values of the objective function to be minimized and its change. To check the effectiveness of the proposed approach, we consider the problem of minimizing the matching error affecting the input information of a fuzzy controller.

II. THE PROPOSED ADAPTATION MECHANISM

A. The fuzzy controller from Nomura et al.

The self-tuning method of fuzzy controllers developed by Nomura et al. is a well-known gradient descent technique to optimize both the fuzzy antecedent and crisp consequent parts. Our objective here is to improve the performance of the gradient descent tuning algorithm by adapting the gradient step iteratively using a set of fuzzy rules to achieve better precision and better convergence rate. This method relies on having a set of input-output data against which the controller is tuned. The FLC consists of a set of n fuzzy rules of the form

Rule i:
$$IF\ x_1\ is\ \ X_1^{(i)}\ and......and\ x_m\ is\ \ X_m^{(i)}\ THEN\ u\ is\ \ U^{(i)}$$

Proceedings 17th European Simulation Multiconference
(c) SCS Europe BVBA, 2003 ISBN 3-936150-25-7

where $x_1,.......,x_m$ are the controller inputs, u is the control output variable, i is the rule number, $X_1^{(i)},.....,X_m^{(i)}$ are linguistic values of the rule-antecedent, $U^{(i)}$ is the linguistic value of the rule-consequent.

The membership functions, $\mu_{X_j}^{(i)}$, of the antecedent part are triangles described by a peak value a_{ij}, and a support b_{ij}, in the defined universe of discourse. The membership function is thus given by

$$\mu_{X_j}^{(i)}(x) = 1 - \frac{2\left|x - a_{ij}\right|}{b_{ij}} \tag{1}$$

The control output membership function is a fuzzy singleton set defined on the real number u_i. Using the max-dot composition and the Center-of-Area defuzzification method, the global control output from the fuzzy rule set is given by

$$u = \frac{\sum_{i=1}^{n} \mu_i u_i}{\sum_{i=1}^{n} \mu_i} \tag{2}$$

where

$$\mu_i = \prod_{j=1}^{m} \mu_{X_j}^{(i)}(x_j) \tag{3}$$

B. The modified gradient descent algorithm

If a reliable set of training data is available that describes the desired control output, u^r, for various values of the process state, $x_1^r, x_2^r,....,x_m^r$, the fuzzy controller can be tuned by minimizing the square error between the FLC output and the desired output given by the reference data. Nomura et al. have chosen to minimize the following objective function

$$J = \frac{1}{2}(u - u^r)^2 \tag{4}$$

Substituting (2) and (3) into (4) gives the following objective function

$$J = \frac{1}{2}\left(\frac{\sum_{i=1}^{n}\left(\prod_{j=1}^{m} \mu_{X_j}^{(i)}(x_j^r) \right) u_i}{\sum_{i=1}^{n}\left(\prod_{j=1}^{m} \mu_{X_j}^{(i)}(x_j^r) \right)} - u^r \right)^2 \tag{5}$$

The steepest descent algorithm seeks to decrease the value of the objective function (5) with each iteration t. In this case, the objective function parameters we wish to alter are the membership function parameters a_{ij}, b_{ij} and u_i. Solving this optimization problem gives the following iterative equations of the parameter values

$$a_{ij}(t+1) = a_{ij}(t) - \lambda_1(t)\frac{\partial J}{\partial a_{ij}}, \quad i = 1,...n; \; j = 1,...m, \tag{6}$$

$$b_{ij}(t+1) = b_{ij}(t) - \lambda_2(t)\frac{\partial J}{\partial b_{ij}}, \quad i = 1,...n; \; j = 1,...m, \tag{7}$$

$$u_i(t+1) = u_i(t) - \lambda_3(t)\frac{\partial J}{\partial u_i}, \quad i = 1,...n. \tag{8}$$

The gradient step updating factor λ_l (l=1,2,3) is calculated using fuzzy rules of the form

Rule k:
IF $J(t)$ *is* $F_1^{(k)}$ *and* $\Delta J(t)$ *is* $F_2^{(k)}$ *THEN* $\lambda_l(t)$ *is* $G_l^{(k)}$

where $J(t)$ and $\Delta J(t)$ are values of the performance criterion and its variation at the iteration t, respectively. $F_1^{(k)}$, $F_2^{(k)}$ are the linguistic values of the rule-antecedent, and $G_l^{(k)}$ is the linguistic value of the rule-consequent.

The functional relationship of λ_l can be viewed as

$$\lambda_l(t) = f(J(t), \Delta J(t)) \tag{9}$$

where f is a nonlinear function (computational algorithm) of J and ΔJ, which is described by a fuzzy rule base.

For determining the fuzzy rule base for computation of λ_l we have taken into account some important considerations related to the optimization problem, i.e., the current values of the objective function to be minimized, the change-of-error and the direction of the gradient vector of the membership function parameters. We attempted to extract IF-THEN fuzzy rules from a linguistic description of a general optimization procedure, that is:

"*we have to look for decreasing the value of the objective function most rapidly in the direction of the negative gradient vector when the current point seems to be significantly far from the desired solution; to slow down the procedure if the current point is close to the solution. If the optimum point is reached the optimization is complete*".

It is very important to note that the rule base for computation of the gradient step will not be dependent on the choice of the rule base for the controller.

C. The Tuning procedure

Once a set of reliable controller input-output data has been collected, a possible optimization procedure is as follows:

Step 1: the rules are fired on the input data ($x_1^r, x_2^r, \ldots, x_m^r$) to obtain the antecedent value μ_i for each rule and the real-valued control output u.

Step 2: gradient step values λ_l are updated using (9).

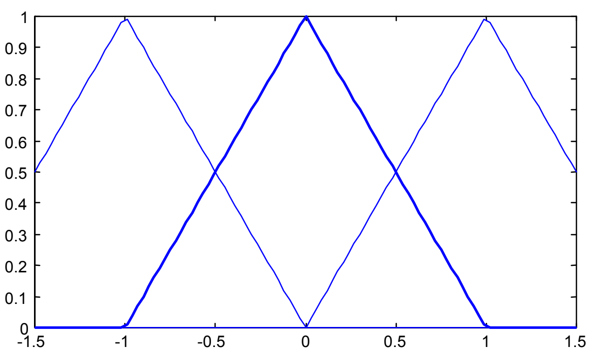

Fig 1. Fuzzy set definition for the input/output variables of the PI-type fuzzy controller

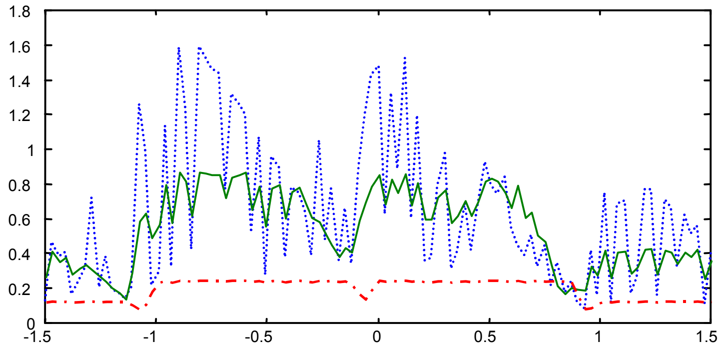

Fig. 3. The matching error for selected values of σ : σ = 025 (dash) ; σ = 1 (solid) ; σ = 3 (dash-dot)

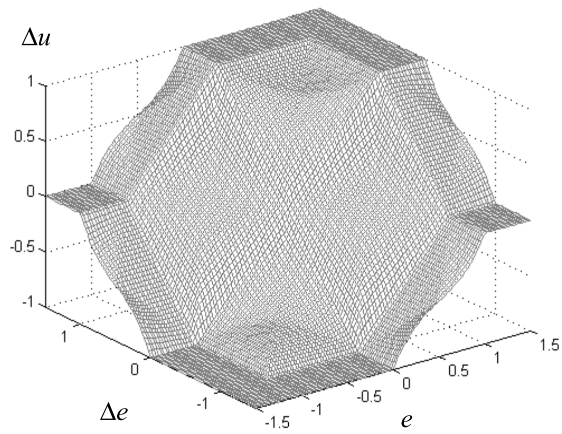

Fig. 2. The control surface of the PI-type fuzzy controller

Step 3: parameters u_i are updated using (8).

Step 4: rule firing is repeated using the new values of u_i .

Step 5: parameters a_{ij} and b_{ij} are updated by (6) and (7), using the values of u_i , μ_i and u.

Step 6: inference error $D(t) = \frac{1}{2}\left(u(t) - u^r\right)^2$ is calculated.

Step 7: if the change-of-error $\left|D(t) - D(t-1)\right|$ is suitably small, the optimization is complete; otherwise it is repeated from step 1.

III. APPLICATION TO THE CONTROL OUTPUT OPTIMIZATION OF THE PI-TYPE FUZZY CONTROLLER

In order to demonstrate the performances of the proposed tuning algorithm, we consider here the optimization problem of a PI-type fuzzy controller. This control problem is solved using both the tuning mechanism suggested by Nomura et al. and the modified version proposed in this paper. The control output of the PI-type FLC is given by

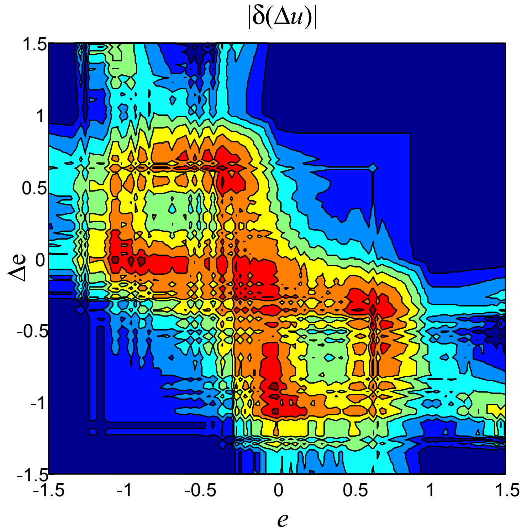

Fig. 4. Contour-plot of the change in the control-output of the PI-type fuzzy controller for $\sigma = 1$

$$u(k) = u(k-1) + \Delta u(k) \tag{10}$$

where k is the sampling instance and $\Delta u(k)$ is the incremental change in controller output. We emphasize here that this accumulation (10) of controller output takes place outside the FLC and is not reflected in the rules themselves. All membership functions for controller inputs, i.e., error (e) and change of error (Δe) and incremental change in controller output (Δu) are triangular-shape partitions uniformly distributed on the common interval [–1.5, 1.5] with three fuzzy set terms: N (negative); Z (zero); P (positive) as depicted in Fig. 1. The PI-type control surface is shown in Fig. 2.

The noise affecting the controller inputs is modeled as an additive random Gaussian variable with a zero mean and standard deviation σ, namely $N(0,\sigma)$. Fuzzy partitions are exposed to controller inputs (e) and (Δe) as well as their noisy versions (e') and ($\Delta e'$). Thus, in fact, the noisy version of (e) induces:

$$\mu_N(e'), \mu_Z(e'), \mu_P(e') \tag{9}$$

instead of the original one:

$$\mu_N(e), \mu_Z(e), \mu_P(e). \tag{11}$$

The matching error is expressed in terms of the overall sum of the absolute differences and is given by

$$r(e) = \left|\mu_N(e) - \mu_N(e')\right| + \left|\mu_Z(e) - \mu_Z(e')\right| + \left|\mu_P(e) - \mu_P(e')\right| \tag{12}$$

The results in terms of $r(e)$ are plotted in Fig. 3. for selected values of σ. The robustness of the control algorithm for which the input fuzzy partition plays the role of an interface is closely related to the input information. Then, any change of the input error, if not absorbed by the fuzzy partition, may have a meaningful effect on the processing error. The difference between the control value (u) obtained for exact input information (e and Δe) and that (u') generated by the controller for the noisy version of the input is illustrated by the contour-plot of the change in the control-output in Fig. 4. In general, this error is viewed as a suitable indicator of fault-tolerance for the fuzzy controller [PED, 93]. In order to optimize the dynamic behavior of the FLC, we propose to use the modified self-tuning mechanism. We choose to only tune the rule-consequent membership functions, via equation (8). The factor which controls how much the crisp consequent values are altered is updated iteratively using equation (9). The centers and the widths of the triangular input fuzzy sets are maintained constant.

A. Performance analysis of the proposed adaptation mechanism

The performances of the proposed tuning mechanism are compared with those obtained by using the original version suggested by Nomura et al. For a clear comparison, we have used some performance measures such as, the final value of the objective function J, and the number of iterations I. This may give a good idea on the precision and the convergence rate of the algorithm with respect to the initial parameters. It can be noticed from table (1) that the proposed tuning mechanism gives the best results in all the cases considered in the simulation study. For example, with a required precision, η, of 10^{-10}, the problem is solved in only 7 iterations using the proposed mechanism; hence it makes 79 iterations with the Nomura's algorithm. In this case, our modified version is some 70 times faster than the simple gradient descent method. In the last simulation case (for $\lambda_0 = 5$), the Nomura's method seems to be inappropriate. The optimization procedure proceeds out of the searching space leading to the divergence of the algorithm. However, the proposed tuning algorithm is much more effective because of the constraints imposed linguistically on the evolution of the gradient step. Using an appropriate architecture (see [GLO, 91]), it will be interesting to implement this mechanism on-line to form an adaptive fuzzy knowledge-based controller.

TABLE I
PERFORMANCE ANALYSIS OF THE MODIFIED TUNING ALGORITHM

Initial parameters		Nomura's method		The proposed method	
λ_0	η	Criterion (J)	Iterations	Criterion (J)	Iterations
0.25	10^{-4}	$9{,}8079.\ 10^{-5}$	25	$4{,}3429.\ 10^{-5}$	02
	10^{-6}	$9{,}3436.\ 10^{-7}$	43	$3{,}8836.\ 10^{-8}$	03
	10^{-10}	$8{,}4799.\ 10^{-11}$	79	$6{,}6947.\ 10^{-11}$	07
0.75	10^{-4}	$4{,}5316.\ 10^{-5}$	08	$2{,}2765.\ 10^{-5}$	02
	10^{-6}	$4{,}9237.\ 10^{-7}$	13	$2{,}0357.\ 10^{-8}$	03
	10^{-10}	$5{,}8124.\ 10^{-11}$	23	$6{,}2062.\ 10^{-11}$	07
1.00	10^{-4}	$8{,}2465.\ 10^{-5}$	05	$1{,}4914.\ 10^{-5}$	02
	10^{-6}	$4{,}0776.\ 10^{-6}$	09	$1{,}3337.\ 10^{-8}$	03
	10^{-10}	$3{,}7596.\ 10^{-11}$	17	$4{,}8688.\ 10^{-11}$	07
5.00	10^{-4}	Divergence		$1{,}0216.\ 10^{-7}$	03
	10^{-6}	Divergence		$1{,}0216.\ 10^{-7}$	03
	10^{-10}	Divergence		$9{,}1359.\ 10^{-11}$	04

IV. CONCLUSION

The problem of designing an adaptive fuzzy controller using gradient descent method has been tackled by proposing an improved self-tuning mechanism. The main contribution of this paper consists of using the approximate reasoning for modeling the optimization strategy with the help of IF-THEN fuzzy rules. The proposed fuzzy rule base is used for the computation of the gradient step which is adjusted continuously with respect to the amount of variation of the performance criterion and the direction of the gradient vector of the fuzzy controller parameters. It has been demonstrated by simulation that the proposed self-tuning mechanism gives more interesting results than a previously published approach. The modified tuning procedure can be used on-line to form an adaptive FLC, if suitable reference data can be generated by considering an appropriate architecture.

REFERENCES

[BOR 90] Borne P., G. Dauphin-Tanguy, J. P. Richard, F. Rotella et I. Zambettakis, *Commande et optimisation des processus.* Méthodes et techniques de l'ingénieur, Editions Hermès, 1990.

[DRI 96] Driankov D., H. Hellendoorn, M. Reinfrank, *An introduction to fuzzy control*. Springer-Verlag, 1996.

[FIS 99] Fischle K., D. Schröder, An improved stable adaptive fuzzy control method, *IEEE Transactons on Fuzzy Systems*, vol. 7, p. 27-40, Feb. 1999.

[GLO 91] Glorennec P.Y., Adaptive fuzzy control,'' *Proc. of the IFSA'91*, p. 33-36, 1991.

[HE 93] He S. Z., S. Tan, F. L. Xu, P. Z. Wang, Fuzzy self-tuning of PID controller, *Fuzzy Sets Syst*., vol. 56, p. 37-46, 1993.

[KIR 98] Kiriakidis K., Fuzzy model-based control of complex plants, *IEEE Transactions on Fuzzy Systems*, vol. 6, p. 517-529, Nov. 1998.

[LEE 93] Lee J., On methods for improving performance of PI-type fuzzy logic controllers, *IEEE Transactions on Fuzzy Systems*, vol. 1, p. 298-301, Nov. 1993.

[NOM 91] Nomura H., I. Hayashi, N. Wakami, A self-tuning method of fuzzy control by descent method, *Proc. of the IFSA'91*, p. 155-158, Brussels, 1991.

[PED 93] Pedrycz W., *Fuzzy control and fuzzy systems*, Second extended edition, RSP Ltd. England, 1993.

[RAJ 99] Rajani K. Mudi, Nikhil R. P., A robust self- tuning scheme for PI- and PD-type fuzzy controllers, *IEEE Transactions on Fuzzy Systems*, vol. 7, p. 2-16, Feb. 1999.

[SAD 75] Sadler D. R., *Numerical methods for nonlinear regression*, St. Lucia, University of Queensland Press, 1975.

[WON 98] Wong C., B. Huang, J. Chen, Rule regulation of indirect adaptive fuzzy controller design, *IEE Proc.- Control theory appl*., vol. 145, p. 513-518, Nov. 1998.

[YOS 90] Yoshida M., Y. Tsutsumi, and T. Ishida, Gain tuning method for design of fuzzy control systems, in *Proc. Int. Conf. Fuzzy Logic Neural Networks*, Japan, p. 405-408, 1990.