NONLINEAR CONTEXT ADAPTATION WITH A GENETIC ALGORITHM

Ricardo R. GUDWIN[a], Fernando A. C. GOMIDE[a], Witold PEDRYCZ[b]

[a] *Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, Departamento de Engenharia de Computação e Automação Industrial, CP 6101 - 13.083-970 - Campinas - SP - Brasil, e-mails: gudwin@dca.fee.unicamp.br, gomide@dca.fee.unicamp.br*

[b] *University of Manitoba, Department of Electrical and Computer Engineering, Winnipeg, Canada R3T 2N2, e-mail:pedrycz@eeserv.ee.umanitoba.ca*

Abstract: In this paper we elaborate on the use of nonlinear functions in the adjustment of membership functions in fuzzy processing. Using such a nonlinear scheme, called nonlinear context adaptation, the knowledge can be encoded in a standard frame of reference, and have its meaning tuned to the situation by means of a nonlinear transformation. Due to its nonlinear characteristic, different parts of base membership functions can be stretched or expanded in order to best fit the desired format. We discuss the use of a genetic algorithm for the discovery of such nonlinear function, given the base membership functions and a set of samples concerning data extracted from environment classified by means of fuzzy concepts.

Keywords: context adaptation, frame of cognition, nonlinear transformation functions, information granularity

1. Introduction

Currently, one of the most challenging areas of investigation concerning fuzzy information processing is the adjustment of membership functions to best represent concepts in real environments. The fine tune of such membership functions is critical when evaluating the effectiveness of fuzzy solutions to engineering problems [1], e.g., fuzzy controllers, fuzzy modeling and fuzzy clustering.

Handling linguistic terms, represented by a family of fuzzy sets, does provide a sort of linguistic space quantization. These linguistic terms play an instrumental role in encoding both numerical and non-numerical information [2].

The main problem is how to find the fuzzy sets that best represent the linguistic terms they are associated to, with the objective of achieving the best model, or an optimum control.

Different attempts have been made in a way to solve this problem.

Earlier approaches tried to conditioning the numerical variables in question, to make them fit into normalized universes of discourse. It was done mainly with the use of gains associated to each variable, implementing a linear scaling of such numerical variables prior to encoding them in terms of fuzzy sets. In essence, these changes do not concern the universe of discourse. The methods of this type were characteristic to fuzzy controllers (scaling factors); they were firstly reported in [2], cf. also [3]. Those gains were initially considered as project parameters. More recently, some sorts of adaptive techniques were used to determine those gains dynamically, through learning [4].

Other methods tried to determine the membership functions by themselves. Some of them do not care for the relation among linguistic terms and the semantic they hold, and generally address fuzzy concepts directly by its membership functions. Examples of such methods are found in [5],[6] and [7].

Some methods use a hybrid approach, adjusting both the gains and membership functions. An example of such approach, applied in fuzzy control, is found in [8], where membership functions of input variables and the gain for the output variable are adjusted.

In [9] it was proposed the idea of linear and polynomial-based modification of the universe of discourse as a way of adjusting membership functions.

In [10], the authors introduced the idea of using contextual information for the assignment of meaning to linguistic terms. The main idea here is that some sort of knowledge is general, and independs from the current context (e.g., that BIG is greater than MEDIUM that is greater than SMALL). This knowledge is represented by means of base membership functions, defined not in terms of any particular universe of discourse, but in a reference one (e.g., the interval [0,1]). For the time of assignment of meaning to a linguistic term, a context is generated (in the studied cases, an interval [a,b]), based in samples collected from environment (many different approaches for doing that are indicated in [10]), and the final membership function is determined by a mapping from the base membership functions scaled to fit the context interval. In [10], only linear context adaptation is analyzed. In [11] we propose an extension for non-linear mappings using neural networks to implement such maps. The great advantage in using a non-linear mapping is that the membership function format can be changed, what is not possible using linear context adaptation.

So, the problem we are concerned with can be posed in two main perspectives. First, as a way of fine-tuning fuzzy systems (controllers or models). But it should also be considered in the way it attempts to answer a key question about generality of linguistic terms, i.e. in how context does its influence in the understanding of concepts.

Next section formally presents the nonlinear context adaptation problem, and in section 3 we propose a way of dealing with it. In section 4 we develop a solution using a genetic algorithm, and in section 5 we describe the performed experiments and the results obtained so far.

2. Problem Statement

Let us consider a collection of fuzzy sets (linguistic terms)

$$A = \{A_1 , A_2 , \dots , A_c \}$$

defined in [0,1]. As usual, cf. [2] we require that A satisfies some obvious requirements of semantic integrity [2,3] such as unimodality and normality of the membership functions of A_i 's.

Given a data set of experimental outcomes (coming e.g. from a certain process of expert polling), they can be arranged in the form of N (c+1)-tuples, namely

$$(d_1 , (\mu_{11} , \mu_{12} , \dots , \mu_{1c}))$$
$$(d_2 , (\mu_{21} , \mu_{22} , \dots , \mu_{2c}))$$
$$\dots$$
$$(d_N , (\mu_{N1} , \mu_{N2} , \dots , \mu_{Nc})) .$$

Our intent is to accommodate these data to the highest extent by adapting the context of A. The essence of this process is to map the unit interval of the "base" universe of discourse to a more suitable universe of discourse, e.g. [a,b] with $a = \min_{i=1\dots N} d_i$, $b = \max_{i=1\dots N} d_i$.

The essence of this context adaptation is that adopting a unitary interval [0,1] as ours base set, [0,1] is mapped onto [a,b] in a nonlinear form so that some of the regions of [0,1] are contracted while others could be expanded. Afterwards, by applying an inverse of this mapping we obtain the family of fuzzy sets

$$A' = \{A'_1 , A'_2 , \dots A'_c \}.$$

It is remarkable that the effect of this nonlinear mapping may result in the modification of the granularity of fuzzy sets of the frame of cognition adapted to this new data.

To properly formalize the problem, let us define the mapping $\varphi : [0,1] \to [a,b]$ in more detail. The formal requirements involve :

(i) **Continuity**

(ii) **Monotonicity** - more precisely we require that φ is nondecreasing (it can be almost constant over some regions). By doing that we assure that the meaning of A_i 's does not change.

(iii) **Boundary Conditions** - the boundary conditions $\varphi(0) = a$ and $\varphi(1) = b$ allow us to accommodate all experimental data.

We can also request that φ is differentiable, which is merely a matter of optimization convenience.

There are a number of potential candidates transforming functions. The one of a particular interest includes a linear combination of sigmoidal functions,

$$u = \varphi(x) = \alpha \,.\, \sum_{i=1}^{c} \frac{k_i}{1+\exp\left(\frac{-(x-m_i)}{s_i}\right)} + \beta$$

where "c" denotes the number of sigmoidal functions with parameters k_i (scale), m_i (translation) and s_i (steepness). We have

$$k_i \geq 0$$
$$m_i \in [a,b]$$
$$s_i > 0$$

The scale factors α and β are computed as:

$$\alpha = \frac{1}{\sum_{i=1}^{c} \frac{k_i}{1+\exp\left(\frac{-(b-m_i)}{s_i}\right)} - \sum_{i=1}^{c} \frac{k_i}{1+\exp\left(\frac{-(a-m_i)}{s_i}\right)}}$$

$$\beta = -\alpha \,.\, \sum_{i=1}^{c} \frac{k_i}{1+\exp\left(\frac{-(a-m_i)}{s_i}\right)}$$

Figure 1 illustrates the underlying idea

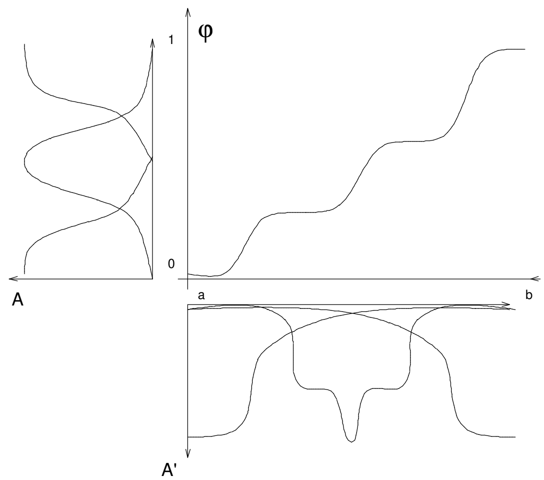

Figure 1 - The Nonlinear Context Transformation

Once φ has been constructed, it is used to generate the new frame of cognition A' whose fuzzy sets are computed as:

$$A'_i(x) = A_i(\varphi(x)),\ x \in [a,b],\ \varphi(x) \in [0,1]\ .$$

3. Computation of the Transforming Function

The determination of φ is completed via optimization computations. For the discussed on-line form of training we assume a pair of data

$$(d_i, (\mu_{i1}, \mu_{i2}, \ldots, \mu_{ic}))\ .$$

Furthermore, considering the MSE (Minimum Square Error) criterion, one gets:

$$Q = \sum_{i=1}^{N}\sum_{j=1}^{c}\left(\mu_{ij} - A(\varphi(d_i))\right)^2$$

Remembering that φ is a function with 3c parameters k_1 , m_1 , s_1 , ... , k_c , m_c , s_c , the determination of φ reduces to:

$$\min_{k_1,m_1,s_1,\ldots,k_c,m_c,s_c} Q \quad \text{s.a.} \begin{cases} k_i \geq 0 \\ m_i \in [a,b] \\ s_i > 0 \end{cases}$$

The resolution of this problem by means of standard optimization computations is very complicated, as it will fall in a system of nonlinear equations. To solve this problem we use, instead, a genetic algorithm.

4. Genetic Algorithm Description

The parameters to be optimized are encoded in a chromosome. The encoding chromosome is generated in the following way. Each gene of the chromosome is a floating point number representing the following sequence of parameters:

$$[\ k_1, m_1, s_1, k_2, m_2, s_2, \ldots, k_c, m_c, s_c\ ]$$

It is generated a random population of such chromosomes, and for each one the value of Q is computed and used as the performance index. Several new populations, each one with the same original number of individuals, are then generated, using this population as a reference:

1) A population generated by the uniform crossover of individuals from the reference population, selected by means of the Roulette Wheel algorithms, based in its performance index [12], and crossing over with a rate of 0.5 of its genes.

2) A population generated by the inductive mutation of the best fitting individual, where each gene of this individual is added to a random value ranging from -50% to +50% of its value.

3) A population generated by the standard mutation operator applied to individuals from the reference population, chosen by the Roulette Wheel algorithm. In this case 50% of its genes are randomly changed to values within the valid range for each gene.

4) A population generated by the inductive mutation of the best fitting individual, where each gene of this individual is added to a random value ranging from -1% to +1% of the valid range for each gene.

5) A population generated by the uniform crossover where one chromosome is taken from the reference population by using the Roulette Wheel algorithm, and the second one is chosen randomly among all the mutated populations 2, 3 and 4. This crossover also uses a rate of 0.5.

After generating those populations, all the individuals are mixed and ranked according to their performance index. After that, the original number of individuals are selected to remain in the next generation. The genetic algorithm evolves by a large number of evolutionary steps. After a given number of steps, the best individual is used to represent the optimized parameters.

5. Experiments

For simulation purposes, we assume the form of φ and membership functions of A (gaussian) are given to generate test data.

In this set of experiments, we arbitrate an input universe of discourse and a nonlinear context function φ as explained in the previous sessions. Then, for equally spaced values x on the chosen universe of discourse, we calculate φ(x) and A_i (φ(x)). The values of x are stored as the samples d_i and the values A_i (φ(x)) are stored as the values μ_{ij} .The samples are then used by the GA, to find the original parameters. Here we consider two candidate functions. The first, F1 (see figure 2), is a strictly monotone function. For this case the genetic algorithm has always converged to the correct solution. For the second, F2 (see figure 4), however, the genetic algorithm always converged to a set of parameters, but sometimes those parameters do not correspond to the correct solution, but to a local minimum. The function shape differed from the original one, but kept consistent with it in the least squares' sense. Therefore, in the case of local minimum, the result is an approximation for the actual transforming function. The linguistic terms defined in [0,1] were characterized by gaussian membership functions of the form $A_i = G(x, m_i, \sigma_i)$ where $G(x,m,\sigma) = \exp(-(x-m)^2 / \sigma^2)$.

For all the experiments, we used the following values:

$a = 5$; $b = 10$; $c = 3$; $N = 20$
$A_1 = G(x,0,0.25)$
$A_2 = G(x,0.5,0.25)$
$A_3 = G(x,1,0.25)$

5.1 First experiment: Function F1

Original Parameters :

K_0:0.30 m_0:5.10 s_0:0.70 K_1:0.40 m_1:5.50
s_1:0.80 K_2:0.50 m_2:6.00 s_2:0.90

Obtained parameters:

K_0:0.02 m_0:6.56 s_0:0.57 K_1:0.01 m_1:5.03
s_1:0.98 K_2:0.32 m_2:5.09 s_2:0.98

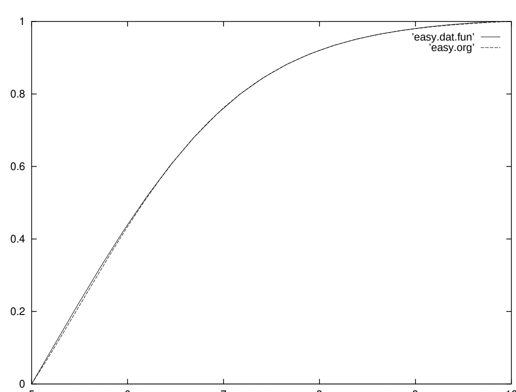

Figure 2 - Function F1 for Experiment 1

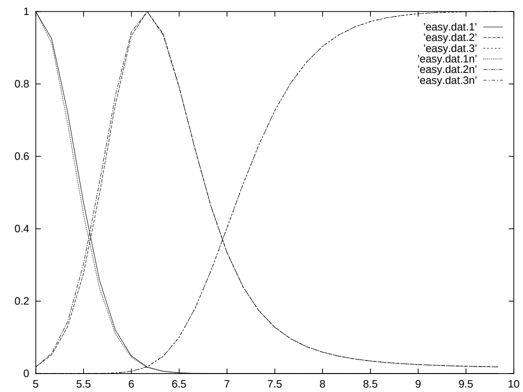

Figure 3 - Membership Functions of A´, Experiment 1

5.2 Second Experiment : Function F2

Original Parameters :

K_0:0.75 m_0:6.25 s_0:0.10 K_1:0.74 m_1:7.50
s_1:0.10 K_2:0.75 m_2:8.75 s_2:0.10

Obtained Parameters: First Trial

K_0:0.48 m_0:6.84 s_0:0.36 K_1:0.22 m_1:8.76
s_1:0.09 K_2:0.01 m_2:8.31 s_2:0.36

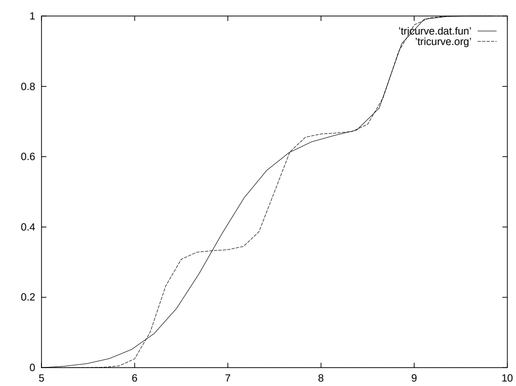

Figure 4 - Function F2 for Experiment 2, Trial 1

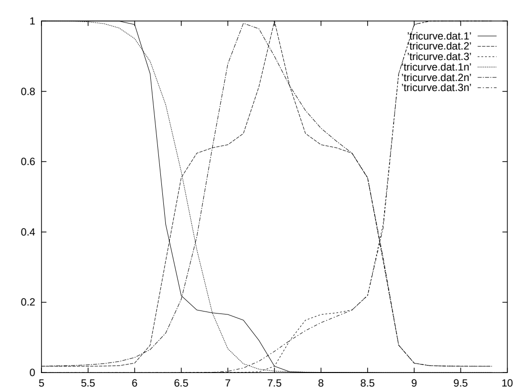

Figure 5 - Membership Functions for Trial 1

Obtained Parameters: Second Trial

K_0:0.00 m_0:6.32 s_0:0.63 K_1:0.22 m_1:6.24
s_1:0.08 K_2:0.63 m_2:8.20 s_2:0.57

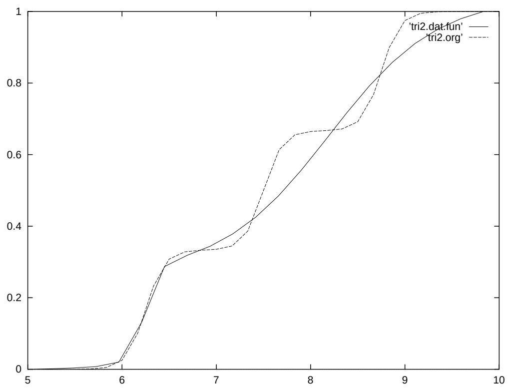

Figure 6 - Function F2 for Experiment 2, Trial 2

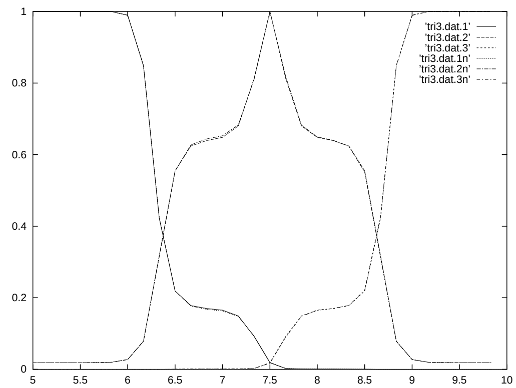

Figure 9 - Membership Functions for Trial 3

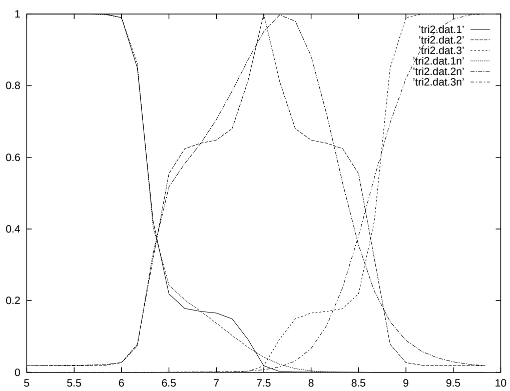

Figure 7 - Membership Functions for Trial 2

Obtained Parameters : Third Trial

K_0:0.75 m_0:6.25 s_0:0.10 K_1:0.74 m_1:7.50 s_1:0.10 K_2:0.75 m_2:8.75 s_2:0.10

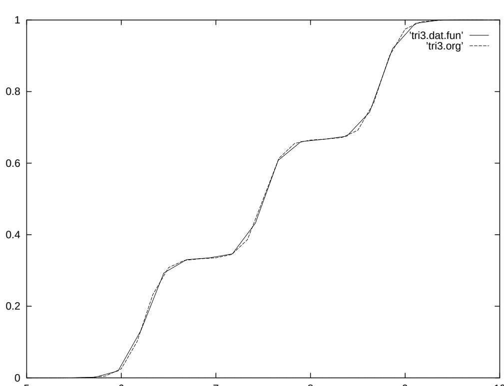

Figure 8 - Function F2 for Experiment 2, Trial 3

6. Conclusions

The primary intent of this paper was to pose the problem of nonlinear context adaptation and to come up with a relevant optimization framework.

The results have shown that, once the requirements on the membership functions hold, for strict monotone transforming functions, the context adaptation scheme proposed always provided the appropriate, correct answers. Additional experiments (not reported here due to obvious reasons) have confirmed this property. It remains to formally proof this statement. This is a task to be pursued in the future. For non-strictly monotone transforming functions, the results have shown that only (reasonable) approximations are guaranteed because the algorithm may stop at a local minimum. However, as usual in practice, result validation should be done with judgment. Of course, this may imply several trials before the solution is found.

The importance of the addressed idea is evident at the conceptual level. By raising the problem of calibrating fuzzy sets we tackled the issue of attaching meaning of generic linguistic terms, that hold even in different contexts.

The direct application aspects have not been discussed. They could be quite easily envisioned in adaptation of rule-based control and fuzzy modeling. These will be analyzed in a separate study.

7. References

1. M. Braae, D.A. Rutherford. "Selection of Parameters for a Fuzzy Logic Controller" - *Fuzzy Sets and Systems*, 2, 185-199, 1979.

2. W. Pedrycz. *Fuzzy Control and Fuzzy Systems*. Research Studies Press/ J. Wiley, Taunton/N. York, 1989.

3. W. Pedrycz. *Fuzzy Sets Engineering*. CRC Press, Boca Raton, FL, 1995.

4. L. Magdalena. "Adapting Gain and Sensibility of FLCs with Genetic Algorithms" - *Proceedings of the 6th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems - IPMU'96* - vol. II, pp. 739-744, Granada, Spain, 1-5 July, 1996.

5. O. Cordón, F. Herrera. "Generating and Selecting Fuzzy Control Rules Using Evolution Strategies and Genetic Algorithms" - *Proceedings of the 6th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems - IPMU'96* - vol. II, pp. 733-738, Granada, Spain, 1-5 July, 1996.

6. F. Herrera, M. Lozano, J.L. Verdegay. "Design of a Control Rules Base Based on Genetic Algorithms" - *Proceedings of the 6th International Fuzzy Systems Association World Congress IFSA'95* - vol. I, pp.265-268, São Paulo, Brazil, 1995.

7. Velasco, L. Magdalena. "Genetic Learning Applied to Fuzzy Rules and Fuzzy Knowledge Bases" - *Proceedings of the 6th International Fuzzy Systems Association World Congress IFSA'95* - vol. I, pp.257-260, São Paulo, Brazil, 1995.

8. Oh, D.J. Park. "Self-Tuning Fuzzy Controller with Variable Universe of Discourse" - *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics* - pp. 2628-2632, Vancouver, Canada, 1995.

9. J.C.T. van der Veen. *Fuzzy Sets, Theoretical Reflections, Application to Ship Steering*. M.Sc. Thesis. Dept. of Electrical Eng., Delft University of Technology, Delft, the Netherlands, 1976.

10. R.R. Gudwin, F.A.C. Gomide. "Context Adaptation in Fuzzy Processing"- *Proceedings of the I Brazil- Japan Joint Symposium in Fuzzy Systems* - pp. 15-20, Campinas - SP, Brazil, July 1994.

11. W. Pedrycz, R.R. Gudwin, F.A.C. Gomide. "Nonlinear Context Adaptation in the Calibration of Fuzzy Sets" - *Fuzzy Sets and Systems* (to appear).

12. D.E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning* - Addison-Wesley Publishing Company, Inc. 1989.

Acknowledgements: The first author is grateful to CNPq, the Brazilian National Research Council,, for fellowship # 360354/96-3 and to FAPESP, the Research Foundation of the State of São Paulo. The second author acknowledges CNPq for grant # 300729/86-3.