Context Adaptation in Fuzzy Processing

Ricardo R. Gudwin   Fernando A.C. Gomide
UNICAMP/FEE/DCA
C.P. 6101
13.081-970 Campinas - SP - Brazil

ABSTRACT: This paper deals with the influence of the context in fuzzy rule base systems. The linguistic variable definition of Zadeh is modified to permit context adaptation, and inference procedure is modified to handle context adaptation when evaluating fuzzy concepts. Procedures to determine context in different situations are provided. Context determination may be viewed as a kind of learning. An application example concerning supervisory control of group of elevators is also considered.

I. Introduction

A linguistic variable is a variable that assigns as its values linguistic labels, which are mapped onto the real world by a fuzzy set. A great problem which comprises this definition is the difficulty to fine tune membership functions, which is actually one of main constraints in designing applications with fuzzy logic. An important matter that have been neglected, is the relevance of context in fuzzy evaluation.

Psychologists have long been intrigued with the finding that context appears to influence perception. The same stimulus information in different contexts can produce different perceptual events. The effect of context in perception has been widely investigated by cognitive psychology. Early studies have been made analyzing context influence in anaphoric references resolution [1], conceptual combination [2] and speech and word recognition [3, 4, 5].

The main idea behind the definition of a context is the idea of restriction. When a context is fixed, what is being made is a restriction of the working universe of a system. In spite of working in the whole universe, the context restricts the system to a particular universe, with its own behavior. The fact of working in a restricted universe modifies perception.

The use of linguistic labels in fuzzy processing is related to concepts. A concept can be understood as a classification rule that defines (fuzzy) subsets. Two kinds of concepts can be identified. The first one is the *absolute* concept. There, the classification rule that defines concept is arbitrary and definitive. The second kind is the *relative* concept. In this case, the classification rule that defines the concept is not arbitrary and definitive, but depends on a comparison relation among a particular element of a set and all other elements on a base set. This base set is used as a reference to classify a particular element in a concept. Examples of such kind of concepts, like *high*, *low*, *medium*, *great*, etc., are widely used in fuzzy rule based systems. This kind of concept is particularly affected by the evaluation context. Depending on our definition of base set, the classification rule that defines the concept will relate a particular element of a set to one or other concept. So, a relative concept will always need a context to be defined, even if this context is the own universe of discourse.

In traditional approaches of concept evaluation by means of linguistic variables, the distinction between absolute and relative concepts is not done. So, when defining a relative concept, the context (which, as we saw, is always necessary) is fixed to the universe of discourse. This kind of representation permits the use of relative concepts in traditional linguistic variables, but it does not explore all the potential of a relative concept. Relative concepts can be defined free of context, being associated to a context only at the time to be evaluated. This permits a single definition to be used in several different contexts, and this way of representation is not explored by traditional linguistic variable approach. Nevertheless we can work with relative concepts in fixed contexts (as in the traditional approach), in some situations the existence of a generic representation of relative concepts, valid to many contexts, more than being handsome, is useful to understand how to apply some adaptive procedures to fine tune the knowledge involved. Generally, in this case, the expert starts from a context which he also knows, the context of his expertise, and then define a fuzzy set to represent a concept, assigning a linguistic label to this fuzzy set. But what happens if we have a changing context ? Or what should be done if we have the same measure type, took in different contexts ? In the first case, adaptive mechanisms may be applied to the original fuzzy sets to fine tune the semantics of its knowledge. In the second case, a measure may be considered using different membership functions, one for each context. Using the idea of contexts and full representation of relative concepts, it will be possible to consider just one membership function on both cases. In the first one, the only need is to redefine the context and the necessary adaptation is just at context definition. In the second case, we will have the same membership functions and consider the measures, one to each context.

II. Modified Linguistic Variable

To permit the processing of context dependent fuzzy concepts, we propose a modification of traditional linguistic variable definition, to enhance its power of representation with relative concepts definition, free of context. This enhancement will permit the application of the same linguistic variable to multiple concepts.

The traditional linguistic variable [6] is characterized by a quintuple (x, T, U, G, S) in which x is the name of the variable; T denotes the term-set of x, that is, the set of linguistic labels used as values of x, with each value being associated to a fuzzy set defined in a universe of discourse U; G is a syntactic rule (which usually has the form of a grammar) for generating composed values of x (values defined as compositions of linguistic labels, logical operators and modifiers like *very*, *fairly*, etc.); S is a semantic rule for associating with each composed value its meaning, which is a fuzzy set of U. A particular value X, generated by G, is called a term.

Our modified linguistic variable will be characterized by a sextuple (x, T, U, G, S, C), which is the traditional quintuple plus a context C. There are other little differences. Depending on the type of concept held by linguistic variable, the use of the universe of discourse U will hold a different meaning. For absolute concepts, the notion of context is meaningless, and we can make it equal to the universe of discourse U. But, for relative concepts, a new meaning is related to U, C and S. In this kind of concepts, the classification rule that define a concept (its semantic) could not be made by means of a fuzzy set defined onto a universe of discourse. Due to its relative meaning, we will need to describe such concepts as fuzzy sets onto base sets, that will be conveniently transformed later, to refer to the particular context under evaluation. So, the element U of the sextuple is not considered as an universe of discourse, but as a base set, over where the semantics S are defined. This special kind of set (base set) could be view as a set where the numeric value of its elements is not important, but what matters is the ordination among its elements. Two frequently used base sets are the finite subset of natural numbers U1 = {0, 1, 2, ... , k} and the unit interval U2 = [0,1]. Two ways of describing the semantics S, in this case, should be by means of discrete fuzzy sets (defined onto U1), or fuzzy sets with extended boundary, defined onto U2.

DEFINITION 1 : Fuzzy Set with Extended Boundary

A fuzzy set is said to be with extended boundary, when its membership functions are defined as:

$$\mu(x) = \begin{cases} g(a), \text{if } x < a \\ g(x), \text{if } x \in [a, b] \\ g(b), \text{if } x > b \end{cases}$$

The context C can be defined then, as a set of the same kind of its base set. For example, if the base set U is a finite subset of naturals, C can be any finite set with the same cardinality of U. If U is the unit interval, C can be any interval. This is done to guarantee that every time a bijective function can be asserted between C and U. This bijective function *Sc*, acts as a scaling transformation, and will be used to assign a context to a concept. To evaluate an element e in a particular context, we take the scaling function from this context to the base set, finding *Sc*(e). This value is then used to find the membership degree of this element, consulting the membership function defined onto U.

The modified linguistic variable has some properties: when learning fuzzy concepts, this learning can be split in two phases; membership functions definition can be done off-line; on-line adaptations are allowed in a second phase learning, since adaptation is just a changing in context; this adaptation saves processing time, once the unique computations will be the evaluation of two parameters (new bounds of context interval); It saves memory, as multiples references to the same concept, in different contexts are related to the same function table, adjusting only the context interval; it preserves the semantics of a relative concept in variable conditions environments.

The main idea behind context separation in modified linguistic variable is the semantics preserving among context instances. By this we mean that the semantics of concepts like "low", "medium" and "high", once defined, remains the same to any context we could refer to. The only thing that changes is the particular context bounds of each context instance. If we analyze human understanding of relative concepts, we would see that when a human being is learning some relative concept, what he is really doing is characterizing a context to the situation in question. The idea behind what is "low", "medium" or "high" is assumed known a priori. What is done is a mapping of the information in a particular context onto those also acquired fuzzy concepts. So, when we refer to concepts like that, and talk about adaptive fine tuning, the adaptation necessary is not at the concepts. The concepts, i.e. the relation among the linguistic terms "low", "medium" and "high" is already fixed mapped onto a base set. What must be adapted, or fine-tuned, is the particular context, i.e., the bounds of variable's domain interval. Such bounds are considered for effective evaluation of the concepts. As we will see in the sequel, both for human being and computer systems, the context interval is constructed by analysis of multiple samples, delimiting the true universe of discourse of a variable, to be considered. We start from a default bounded interval and then translates the bounds, as new knowledge is incorporated to the system. An expert has a refined knowledge, because his expertise was developed by all sample cases he considered during his learning experience.

III. Context Determination Procedures

We have basically two ways of generating a context. The first one is by static comparisons. In this case, we compare multiples instances related to a concept to create a context. When we do not have multiples instances related to a concept, we must do temporal comparisons, by the same variable analyzed. Those two kinds of comparisons, lead to the two types of context determination procedures.

A. Simultaneous Samples Procedures:

If we have simultaneous instances of the same kind of variable, we can compare them each other to determine a context. An example of this kind of context determination could be the number of persons at an elevator hall, in a given time. As we have a hall for each floor of a building, we can compare the number of persons in a given hall to all other halls in a given time. This comparison will permit us classify a particular hall density of people as "low", "medium", or "high", as the context under evaluation is considering multiples instances of the variable "number of persons in hall" at the same time. This is useful to assigning priorities for hall calls. When we do comparisons like that, we are using simultaneous samples procedures.

B. Temporal Samples Procedures:

When we do not have multiples instances of a variable, like the number of persons in hall, we must compare instances temporally distributed, to determine a context. In this case, we must take samples during a pre-defined slice of time, and then do a statistic evaluation of those samples. This statistic will then determine a context.

This kind of context generation is similar to human context generation. Take the following example:

Consider a variable which you do not have any information about, as the concentration of a reagent in a chemical process. If it is given a first information that the concentration of this reagent, in a first sample is 3 ppm, one cannot classify this information as "low", "medium" or "high" (any kind of classification will be using a pre-defined context of a similar variable). But, if we say that last sample measured 18 ppm, one is induced to first classify the new sample as "low", as a particular context is becoming to be developed by given information. Again, if we say that most samples measure about 3 ppm, we can reevaluate our classification and say that the first sample concentration of a reagent is "medium", as the context definition is becoming more "solid". This is an illustrative example of how a temporal context is generated. Following, we will analyze some basic context generation procedures and suggest other possibilities derived from them, allowing adaptive procedures for context determination.

IV. Examples of Procedures

A. Absolute Limit Context Determination:

To define a context, we must take samples and generate the upper and lower bounds of an interval. In absolute limit method, the lower bound is just the lower instance of sample set, and upper bound is the highest instance of sample set. For example, in the elevator group control problem, consider the variable "call waiting time", which measures the time a call is waiting at a floor, in a building. In this case, we can do a comparison among simultaneous samples: all the calls in the building. Suppose that waiting time of those calls are the following:

{10, 21, 17, 15, 12}

So, this set is our sample set, and the absolute limit method will define a lower bound LB = 10 and an upper bound UB = 21. This interval defines a context to the given time. The main property of this method is its simplicity. This method is very simple to be implemented and functions very well, when samples are well distributed. Some problems could occur when samples are not well distributed. Consider that instead those waiting times, we have the following:

{51, 1, 55, 57, 53}

If we use absolute limit method, the subtle differences among the four greatest values could be bad evaluated. To define the sample set, in the case of simultaneous samples comparisons, the sample set dimension is the multiplicity of variable's instances. When doing temporal comparisons, it is necessary to define a time-slice where samples are collected, and after taking all samples, the method can be applied. More sophisticated sample set definition could be the recording of a number of past states, in a FIFO buffer, and using those recordings as the sample set, or doing averages of past states and recording them in FIFO buffers, generating different orders of time magnitude for temporal evaluation. (Something like the context of last 10 minutes and the context of last 24 hours).

B. Elastic Limit Context Determination:

The elastic limit context determination can be used both by simultaneous samples or by temporal samples. This context determination procedure adapts continuously the bounds of context interval, by an exponential filter. Given a sample, the method first determines which bounds will be adapted. If sample (S) > upper bound (UB), then only UB is adapted. If S < lower bound (LB), then only LB is adapted. If LB < S < UB, both LB and UB are adapted. The law of adaptation is the following:

$$\text{NewBound} = \alpha \cdot (\text{Old Bound}) + (1-\alpha) \cdot \text{Sample}$$
$$0 \le \alpha \le 1$$

The parameter α will determine the velocity of convergence. As closer α is from 1, more slowly the bound will change. As closer to 0, more fast the adaptation is done. The determination of the best α will be a project parameter. In some cases, it can even be variable.

C. Statistic Context Determination:

The statistic context determination tries to consider the problem where absolute limits seems to fail, i.e., when the distribution of samples are not well suited. This method, uses an statistic evaluation of sample, to define its bounds. By this aspect, this method is not unique, but is a family of methods. The simplest one is the mean and standard deviation method, which calculates the mean and the standard deviation of sample, and assuming that in a normal distribution about 96% of samples are between 3 standard deviations of mean, up and down, it assigns the lower bound to mean minus 3 standard deviations and the upper bound to mean plus 3 standard deviations:

$$M = \sum_{i=1}^{N} \frac{x_i}{N}$$

$$SD = \left(\sum_{i=1}^{N} \frac{(x_i - M)^2}{N} \right)^{1/2}$$

$$LB = M - 3SD$$
$$UB = M + 3SD$$

This method demands more processing time than the absolute limit method, but it does a more adequate treatment to distributed samples. Other statistic measures could be used, like a modal or median analysis.

D. Neural Networks Context Generation:

Using separate context representation with neural networks, a neuro-fuzzy [7,8] scheme of processing can be easily implemented. The main improvement given by this approach is the learning of context by neural network adaptation procedures. Other important improvement is that using neural networks, we should use a discrete framework, opposed to interval contexts used in standard fuzzy processing. Below, we propose a kind of neural network, with a learning rule similar to Kohonen's [9] law of adaptation. This net reflects in its weights, the statistical distribution of inputs. This kind of neural network, is well suited for applications which considers context information. In this case, context is a finite set, where each element of context is an output of a neuron. The input to fuzzy system will not be a single value, as usual, but a fuzzy set in strict sense (a fuzzy singleton), defined as the values of output layer of the neural network. This property permits better integration with other sub-nets, that considers fuzzy processing. In figure 1 we show the main scheme of this neural network.

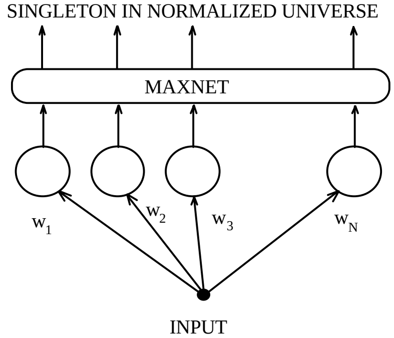

FIGURE 1 - Example of Neural Network

The single input links to each neuron by a weight w_i. Each neuron is adapted by a Kohonen learning procedure, as for example, the one described in [10]. After a significant number of samples (inputs) were presented, the neural network will have as weights, an ordered range of numbers proportional to statistical sample distribution. In figure 1, the MAXNET subnet is the one outlined in [10], where only the neuron with the closest value to input will remain active, all other will be innative. Outputs of MAXNET could then be used by fuzzy procedure to do the matching phase. Despite this feature, it is necessary to detach that as weights are updated closer to sample distribution, its evolution in time corresponds to a context adaptation. So, if membership factors are associated to the weights that link MAXNET output nodes to a superior sub-network, this sub-network could implement the matching phase, with matches the input singleton to a particular fuzzy concept being evaluated.

V. Application Example

In supervisory control of elevator systems the main task is to coordinate the movement of a group of elevators by means of a set of logical rules with the aim of improving the elevator system performance.

The main requirements of a group control system in serving both car and hall calls should be: to provide even service to every floor in a building; to minimize the time spent by passengers waiting for service; to minimize the time took by passengers to move from one floor to another; to serve as many passengers as possible in a given time. In [11], a fuzzy knowledge based group supervisory control of elevator systems was developed. The use of fuzzy knowledge processing is particularly useful because of the following: vague concepts are used to derive allocation strategy; the number of necessary fuzzy rules, compared to conventional logic approaches are deep smaller; multi-objective constraints can be achieved. The particular fuzzy strategy (which is a simplified strategy of a real case), is described in [11]. It considers the current waiting time and the attending time of each call. With those linguistic variables, the rules assigns linguistic priorities to each call, relative to each elevator. This rule base is a typical example of how separate context information being processed can be used to improve systems performance. As stated in [11], membership functions must be tuned to each traffic conditions, in order to achieve the best performance. The context approach was applied to the system developed in [11], using simultaneous samples comparisons for current waiting time and attending time. The sample set was the set of pendent calls (calls which are not allocated), and it was used the absolute limit method and the mean and standard deviation method. The results of simulation, comprising average waiting time, are given at table 1. For standard fuzzy strategy, it was used standard fuzzy processing well tuned, by trial an error, for each traffic pattern.

TRAFFIC CONDITIONS	ALLOCATION STRATEGY		
	FUZZY	ABS. LIMITS	MEAN & SD
off-peak	22s	21s	18s
up-peak	69s	70s	70s
down-peak	44s	43s	44s

Table 1 - Simulation Results - Average Waiting Time

As can be observed by table 1, the modified approach, with simple context determination strategies can achieve practically the same result of well tuned standard fuzzy strategies. This is a great enhancement, as membership functions tuning is one of the main problems in fuzzy programming.

VI. Conclusions

In this paper, the idea of separate context processing from main fuzzy computations were presented, by means of a modification in linguistic variable structure, and its enhancements to fuzzy programming were shown. It divides learning fuzzy concepts into two steps. The first is the learning of fuzzy concepts in a normalized universe, which represents linguistic labels relationship among each other. The second is the learning of the context where those linguistic labels should be applied. Once defined the primary fuzzy sets, they can be applied to any context given with the same type of normalized universe. The primary learning is generally the one that requires more computation effort. This can be done off-line or can be given directly by experts. The fine-tune, which must be done with on-line procedures (in real-time control/decision systems), can be achieved by context adaptation. In this case, our methods are simple and well suited to real-time processing. As fine-tuning is at present time the greatest difficulty in building fuzzy applications, the use of context-based fuzzy processing seems to be of great help to enhance such implementations. The perspective of using neural-networks to generate contexts is somewhat interesting, as single inputs can be converted to whole fuzzy sets, and those fuzzy sets are automatically adjusted to statistical samples from real world.

Acknowledgments: R. Gudwin acknowledges CNPq - The Brazilian National Research Council for a fellowship; F. Gomide acknowledges CNPq for grant #300729/86-3

VII. References

[1] Roth, E.M.;Shoben, E.J. "The Effect of Context on the Structure of Categories" Congnitive Psychology **15**, 346-378 (1983).

[2] Medin, D.L.;Shoben, E.J. "Context and Structure in Conceptual Combination" Cognitive Psychology **20**, 158-190 (1988).

[3] Massaro, D.W. - "Testing between the TRACE Model and the Fuzzy Logical Model of Speech Perception" - Cognitive Psychology **21**, 398-421 (1989).

[4] McClelland, J.L. - "Stochastic interactive Processes and the Effect of Context on Perception" - Cognitive Psychology **23**, 1-44 (1991).

[5] Massaro, D.W. - "Integration versus Interactive Activation: The Joint Influence of Stimulus and Context in Perception" - Cognitive Psychology **23**, 558-614 (1991).

[6] Zimmermann, H.J. - "Modelling Flexibility, Vagueness and Uncertainty in Operatios Research"- Investigación Operativa - vol 1, n° 1, August 1988.

[7] Horikawa S. et.al - "Composition Methods of Fuzzy Neural Networks" - 1990 IEEE Public #087942-600-4/90/1100-1253

[8] Lin,C. - "Neural Network-Based Fuzzy Logic Control and Decision System" - IEEE Trans. on Computers vol 40 n12 Dec.1991

[9] Kohonen,T. "Self-Organization and Associative Memory" Berlin, Germany: Springer-Verlag (1988).

[10] Lippman,R.P. "An Introduction To Computing with Neural Nets" IEEE ASSP Magazine April 1987.

[11] Gudwin, R.R. - "A Kernel for Real-Time Knowledge Processing" - M.S. Thesis, Campinas State University - UNICAMP (1992) (in portuguese)