



**HAL**  
open science

# Fuzzy Programming: Why and How? Some Hints and Examples

Henri Prade

► **To cite this version:**

Henri Prade. Fuzzy Programming: Why and How? Some Hints and Examples. Bulletin pour les sous-ensembles flous et leurs applications, 1980, 5, pp.76–89. hal-04564625

**HAL Id: hal-04564625**

**<https://hal.science/hal-04564625>**

Submitted on 30 Apr 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

- SOME HINTS AND EXAMPLES - (\*)

## I - INTRODUCTION

The procedures used by the human mind, as the data manipulated by these procedures are not always precisely specified. This situation is not generally due to an intrinsic imprecision of the natural language we used : English or French, for instance, are certainly able to express things in a precise way most of the times. It is not generally for the pleasure that people may be imprecise or vague in stating facts or methods, although some of them take advantage of it when they want to hide something ! It is not even due to some intrinsic inability of the mind to put things into words. The deep reason of possible imprecision in specifying seems to be elsewhere. Sometimes precise data are simply not available because we have not the tools to perform precise measures or because it is impossible to do it : the processing time of an operation which has not been performed yet can be only evaluated on the basis of our apriori knowledge : this evaluation is more or less precise according to the situations ; we have a possibility distribution concerning the value of the processing time. When the information is stored in our memory, we may be somewhat uncertain if we try to remember too precise facts. Other times, precision is not meaningful, especially in the human affairs ; for example, the required profile of a candidate to an employment are linguistically specified, even if the elements of this profile can be more or less evaluated or graded in a numerical way. Although we may assess arbitrary values to what is only roughly specified, or we may refuse to consider what is imprecise, it seems more natural and useful to accept imprecise data or procedures and try to deal with them as such, because imprecision or fuzziness is often an unavoidable feature in humanistic systems. Moreover, Zadeh (1973) has pointed out that human mind often uses fuzzy labels in reasoning processes : "Thus, the ability to manipulate fuzzy sets and the consequent summarizing capability constitute one of the most important assets of the human mind as well as a fundamental characteristic that distinguishes human intelligence from the type of machine intelligence that is embodied in present-day digital computers". There is a balance between the imprecision of a procedure and its adaptability to a variety of situations.

(\*) Presented at the Invited Session 'Computer Applications of Fuzzy Set Theory' of the 4<sup>th</sup> IEEE Int. Computer Software & Applications Conference. Chicago, October 27-31, 1980.

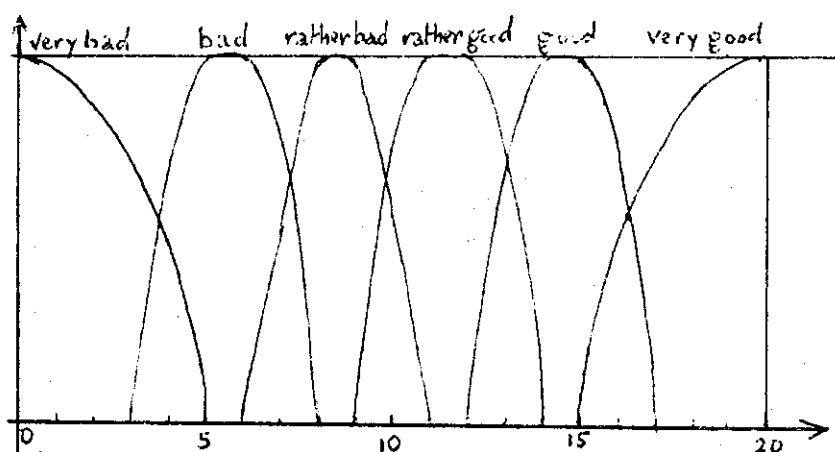
Let us give two examples of fuzzy procedures to which it is referred in the following. First, a fuzzy decision procedure concerning selection of students: The considered examination criteria are for instance : Mathematics, English and French. For each of them, the global evaluation is built as a weighted mean of the grades got by the student during the past year. These grades are supposed to belong to the set of six levels A, B, C, D, E, F which may be linguistically interpreted respectively as 'very good' , 'good' , 'rather good' , 'rather bad' , 'bad' , 'very bad' (This kind of grading has been actually used in French schools). What is required to be selected is for instance "to be at least good in mathematics and to be at least rather good in French or English". The second example is a fuzzy guidance procedure, such that : "Go about 300 meters towards North ; you will see a narrow street ; you will take it and almost immediately you will see a bank". Note that in this example the success of the procedure is guaranteed (if backtracking is allowed) by the existence of a landmark (i.e.the bank) which enables to control (in the sense of automatics) the position.

In the following, we first consider the various kinds of fuzzy instructions (we are not concerned here with the understanding and the translation (into fuzzy instructions) of imperatives expressed in natural language). A semantic pattern-matching procedure taking into account the uncertainty in meaning is then briefly sketched and its use is emphasized. Finally, preliminary discussions about fuzzy data types are presented.

## 11 - FUZZY INSTRUCTIONS

An instruction may be fuzzy because of the presence of fuzzy arguments (i.e. arguments whose values are fuzzy entities), or of fuzzy functions (i.e. functions whose values are fuzzy entities whatever their arguments) or of fuzzy predicates (i.e. predicates yielding truth-values other than true or false). An instruction may be also fuzzy because its arguments are fuzzily designated (even if the value of each possible argument is non-fuzzy).

Fuzzy arguments are, for instance, fuzzy sets on which are performed adapted versions of set-theoretic operations (Umano et al. 1978). Another important type of fuzzy data is 'fuzzy number'. For example, in the selection procedure stated above, we have to compute the weighted mean of grades which are fuzzy sets of the real interval  $[0,20]$ , i.e. fuzzy numbers such the ones whose membership functions are pictured below



Owing to a parametered representation of the membership functions, arithmetic operations extended to fuzzy numbers are easy to perform (Dubois & Prade 1980). Note that what we get is a parametered representation of a fuzzy set, not its semantic label ; if we want a linguistic output, it is necessary to have a linguistic approximation function for labelling fuzzy sets of a given universe of discourse (Bonissone 1978, Eshragh & Mamdani 1979, Mandic & Mamdani 1980). The arithmetic operations performed on fuzzy numbers are not fuzzy functions because when restricted to ordinary numbers, they yield the usual results.

A fuzzy function such as "INCREASE\_SLIGHTLY" may be defined via usual arithmetic operations and fuzzy numbers (e.g.  $x \rightarrow x \oplus \text{EPS}$  where EPS is the label of a fuzzy number whose possible values are small) or directly by <sup>an</sup> analytical function mapping  $x \in \mathbb{R}$  on a fuzzy number (then, this function can be canonically extended to include fuzzy numbers in its domain).

Conditional instructions, which control the sequencing, may have premises (i.e. the "if part" of the instructions) which are fuzzy because of two reasons :

- the predicate is fuzzy (e.g. LARGE , APPROXIMATELY\_EQUAL)
- the values of the arguments are fuzzy, because they result from the execution of fuzzy instructions belonging to the kinds studied above.

For instance, in the selection procedure, we may add : "if the candidate is YOUNG, then, INCREASE\_SLIGHTLY his global grade in mathematics". But, YOUNG is a fuzzy predicate modeled by a fuzzy set on the universe of ages, therefore the consistency between the age of a considered candidate and YOUNG may be intermediary between 0 and 1 ; in other words, it is neither completely true that the candidate is YOUNG nor completely false.

The height of the intersection of the fuzzy or non-fuzzy set modeling the value of the argument with the fuzzy or non-fuzzy set modeling the predicate will

be interpreted as the truth-value of the premise (consistency degree of the values of the argument and of the predicate). This truth-value will be denoted by  $h(\text{AR} \cap \text{PRE})$ . AR and PRE respectively label the fuzzy set of possible values of the argument and the fuzzy set of the elements satisfying the predicate.

A similar procedure can be used to determine the truth-value of the premise when the predicate is n-ary (Dubois & Prade 1980, Adamo 1980). After the evaluation of the truth-value of the premise two actions are possible : either execute the 'then part' of the instruction or go to the following instruction. A truth-value equal to  $h(\text{AR} \cap \text{PRE})$  is associated to the first action. The quantity  $h(\text{AR} \cap \overline{\text{PRE}})$  is associated to the second one ;  $\overline{\text{PRE}}$  has a membership function which is deduced from this of PRE by complementation to 1.

Generally we have :

$$h(\text{AR} \cap \overline{\text{PRE}}) \neq 1 - h(\text{AR} \cap \text{PRE})$$

The formula  $h(\text{AR} \cap \overline{\text{PRE}}) = 1 - h(\text{AR} \cap \text{PRE})$  holds if the argument has a non-fuzzy value.

If we concurrently consider both so-valued actions, we can develop a graph of possibilities which represents a non-deterministic program, whose halting conditions are difficult to determine (Floyd 1967). Practically we are led to use the rule of the preponderant term of the alternative (Zadeh 1973) : only the action corresponding to the greater truth-value is executed.

However, in case where the different branches of the control structure correspond to different modifications of the same set of variables and have the same sequencing after, it is possible to execute one intermediary instruction, which uses the conditional statements as a table giving a sampling of fuzzy points of a function, in order to "extrapolate" the result according to Zadeh's theory of approximate reasoning (1978). For instance, it would lead to modulate the slight increasing of the grade in mathematics according to the age of the candidate in the above example.

In the preceding, the manipulated fuzzy entities were non-fuzzily designated. When, entities are reachable by a fuzzy designation it may exist some ambiguity regarding the addressed entity, if its designation is too fuzzy in the context.

Examples :       LIST-ALL BIG OBJECTS  
                   ADD ( WEIGHT BIG BOX ) ( WEIGHT LITTLE BOX )

In the first example, the system has to find out all the more or less big objects recorded in its associative memory. In the second example, the system has to find out the box whose qualification by BIG has a maximal meaningfulness, (Goguen 1976), the box whose qualification by LITTLE has a maximal meaningfulness and finally to add their weights. In both cases, a procedure of fuzzy pattern-matching (see next section) is needed.

Note that the result, is non-fuzzy as soon as, when the designation is ambiguous in the context, the fuzzy pattern-matching procedure indicates it and stops the evaluation. Another kind of execution will lead to list all the objects with non-zero membership grade in the fuzzy set BIG, with the corresponding grades, for the first example (Le Faivre 1974). A similar execution of the second instruction would yield the possibility distribution of the weights we can obtain !

Especially in robotics, in order to execute fuzzy instructions we have to interpret them in a non-fuzzy way. This "defuzzification" is now imposed by the confrontation with the real world. The execution may remain fuzzy if the real conditions of the execution are ignored.

Examples :

- ( $\alpha$ ) do some steps on the left
- ( $\beta$ ) increase by x the pressure (x is an identifier of the already computed fuzzy quantity)
- ( $\gamma$ ) go-and-fetch the greenish big cub.

In examples ( $\alpha$ ) and ( $\gamma$ ) where the occurrences must have discrete values (the numbers of cubs or steps are integers) a fuzzy instruction is defuzzified by using the already cited maximum meaningfulness principle : from the possibility distribution resulting from the evaluation of the fuzzy elements in the instruction, we keep among the really existing occurrences those which correspond to a maximal possibility. In the example ( $\beta$ ) we directly take a pressure value which corresponds to the maximum of the possibility distribution associated with the fuzzy quantity x, because the pressure can be continuously assigned. The second example of the introduction sketches a robotics-like sequence of fuzzy instructions. Note that in the execution of an instruction such that "go about 300 meters", a demon (in the sense of Artificial Intelligence) should be activated in order to indicate when begins and ends the area where it is more or less possible to find the narrow street the robot is walking for. (This area is defined by the support of the fuzzy set "about 300 meters" : the set of all distances with non-zero membership grades).

In robotics, after Zadeh's (1973) and Goguen's (1975) hints, Smith (1975) designed, in the framework of JASON project, a plan generator fitted to an uncertain environment and ill-stated problems. In the same spirit (Tanaka & Mizumoto 1975, Uragami et al. 1976) have dealt with the control of robots by means of fuzzy instructions. Shaket's system (1976) allows fuzzy designations of objects in a Winogradian world of blocks. Fuzzy hints are used by Gershman (1976) in order to guide a robot in a maze.

In some situations, results have to be defuzzified in order to be used. But in other situations fuzzy results can not only be accepted but even wanted. Indeed, these fuzzy results are a powerful method to represent and communicate information in a human-like way.

Examples of fuzzy results got after linguistic approximation :

- . x is very large
- . the temperature is rather small.

If the results are expressed in a procedural way, they can be viewed as fuzzy instructions. Such fuzzy instructions may constitute valuable plans for a robot. Using fuzzy designations is a way to sum up the useful information; the recording of fuzzy plans may be a powerful way of planning.

### III - FUZZY PATTERN MATCHING

In the following a brief account of a pattern matching system developed by Cayrol, Farreny & Prade (1980) is given. An example illustrates it.

The current pattern-matching procedures are based on a rigid conception of the similitude between the pattern and the datum under consideration. Comparing patterns and data, ordinary methods decide if the datum is acceptable by the pattern. Patterns and data are represented as nested lists of atoms. The similarity between an atom of the datum and an atom of the pattern is considered as a two-valued logical variable, even if this similarity is calculated through more complex functions than the identity. The global similarity between the pattern and a datum is computed as a logical combination of atomic similarities.

In knowledge-based systems it is very convenient to use words of a natural language as atoms.

This system is an attempt to make more flexible the associative access by taking into account the imprecisions due to the fuzziness of the words of natural language.

A word of a natural language may often designate a collection of objects, rather than a unique object ; moreover this collection has generally no crisp boundary : it is more or less possible that an object of the collection is the object actually described. This fuzzy set can be viewed as the meaning of the word (Zadeh).

The membership function of the set is viewed as a possibility distribution.

In this approach a pattern is a tree-like structure whose leaves are "atomic pattern". A possibility distribution is associated with each of them. In the same way the data are represented as tree like structures whose leaves are "atomic data", also associated with possibility distributions.

Two measures of the similarity between the atomic pattern and an homologous atomic datum (homologous in the sense of the morphism between the structures) are evaluated :

- The possibility (Zadeh) that what is designated in the datum is (at least approximately) equal to what is required by the pattern. If  $\mu_P$  and  $\mu_D$  denote, respectively, the possibility distributions attached to an atomic pattern and an atomic datum, the possibility measure is taken to be equal to :

$$\Pi(P \circ R | D) = \sup_{u \in U, v \in U} \min (\mu_P(u), \mu_D(v), \mu_R(u, v))$$

where  $U$  is the universe of discourse, and  $\mu_R$  is the membership function of a relation of (possibly approximate) equality.  $\mu_R$  models a tolerance. In the following example  $\mu_R(u, v) = \max (0, 1 - |u - v| / \lambda)$  with  $\lambda \geq 0$ .



- The necessity  $N(P, R | D)$  of the same event. This concept of necessity is a dual concept of the one of possibility in the sense that:  $N(P, R | D) = 1 - \Pi(\overline{P}, \overline{R} | D)$ ; the necessity of an event is equal to "the impossibility of the opposite event" which yields : 
$$N(P, R | D) = \inf_{v \in U} \max_{u \in U} (\sup_{u \in U} \min(\mu_P(u), \mu_R(u, v)), 1 - \mu_D(v))$$

These atomic measures of possibility and necessity are aggregated in order to yield global measures between the whole pattern and the whole datum. This aggregation (making use of "min" operator) preserves the respective semantics of possibility and necessity of the measures.

Both similarity measures are computed between the pattern and each datum present in the data basis. The data which have the best similarity measures with respect to the pattern are yielded by the system. Several situations may happen :

- No datum sufficiently matches the pattern : the similarity measures are below some thresholds.
- One or several data match the pattern in the same way ; there may be several data which have (at least approximately) the same possibility measure and the same necessity with respect to the pattern ; then the data are said to be interchangeable.
- Several non-interchangeable data are yielded : two data may be sufficiently similar to the pattern, but one of them has the best possibility measure while the other has the best necessity measure. In this case, interchangeable data are gathered in a same cluster ; clusters are incomparable from a choice point of view.

The system described above is implemented in LISP and is currently experimented. This system may be particularly useful in the framework of the communication in natural language with robots, where approximate descriptions of real world situations and approximatively specified rule are needed.

The following example deals with (more or less approximate) description of people. The data available in the considered data base are :

D1: (JOHN THIRTY:AGE TALL:HEIGHT AROUND-SIXTY:WEIGHT)  
 D2: (PETER ABOUT-FIFTY:AGE MEDIUM:HEIGHT OVER-SEVENTY:WEIGHT)  
 D3: (PAUL ABOUT-FORTY:AGE RATHER-TALL:HEIGHT ABOUT-FIFTY:WEIGHT)  
 D4: (TOM RATHER-YOUNG:AGE SMALL:HEIGHT FIFTY:WEIGHT)  
 D5: (RICHARD ABOUT-FORTY-FIVE:AGE TALL-HEIGHT ABOUT-SEVENTY-FIVE:WEIGHT)  
 D6: (HENRY AROUND-FORTY:AGE VERY-SMALL:HEIGHT AROUND-SIXTY:WEIGHT) . ;  
 D7: (ROBERT ABOUT-FORTY-FIVE:AGE MEDIUM:HEIGHT ABOUT-SEVENTY-FIVE:WEIGHT)  
 D8: (PATRICK AROUND-FORTY:AGE MORE-OR-LESS-MEDIUM:HEIGHT ABOUT-SEVENTY-FIVE:WEIGHT)  
 D9: (DAVID AROUND-FORTY:AGE MEDIUM:HEIGHT SIXTY-NINE-AND-HALF:WEIGHT)

The pattern is P : (>X AROUND-FORTY:AGE MEDIUM:HEIGHT OVER-SEVENTY:WEIGHT)

The representation (by 4-tuples) used in the examples are given in appendix . With a tolerance equal to 0 ( $\lambda = 0$ ) for each universe the pattern-matching procedure yields as acceptable (because their similarity measures are above the thresholds) the data D7, D8, D9. The data D7 and D8 are interchangeable : the difference between the homologous similarity measures is below 0.05. More precisely we have :

$$\begin{array}{lll} \Pi(P|D7) = 1 & \Pi(P|D8) = 1 & \Pi(P|D9) = 0.75 \\ N(P|D7) = 0.42 & N(P|D8) = 0.41 & N(P|D9) = 0.5 \end{array}$$

We observe that D7 and D9 (or D8 and D9) are not interchangeable.

The effect of an increasing of the tolerance parameters of the universe of heights is detailed in appendix . In this example, it does not lead to select more data. For sure it is not always the case.

Note that the results are coherent with our intuition. The system yields the data that may correspond to the pattern.

The selection procedure sketched in the introduction, once computed the global profile of each candidate, obviously corresponds to a fuzzy pattern-matching problem.

A little more fuzzy, is the concept of fuzzy rejection. It corresponds to the ordinary functions equipped with a fuzzy domain A (see Dubois & Prade 1980).

IV - TOWARDS FUZZY DATA TYPES

A data type consists of :

- a set of objects, called its members
- a set of functions, called its primitive operations whose domains and ranges are specified. At least either the domain or the range of each primitive operation must include (possibly within some Cartesian product) the own set of members of the data type. The primitive operations operate on or produce members of the data type.
- a set of axioms which describe the way the primitive operations interact with one another, and perhaps with other functions.

The concept of data type has been extensively discussed by Liskov & Zilles, Guttag et al., Parnas et al. . .

Moreover, each data type has, as a member, an ideal element, namely REJECT which is the output of any operation when it has no genuine output of the expected sort. (i.e. when some members of the domain of the operation are not mapped onto members of the range).

After this brief recall, we may try to introduce what is a fuzzy data type (see also Prade et al. 1978).

Naturally, there are several kinds of fuzzy data types more or less fuzzy, depending on where fuzziness lies.

V - CONCLUDING REMARKS

First, only the members of the data-type may be fuzzy entities. But the primitive and derived operations are ordinary ones. The axioms are also of a classical kind. Thus, we have for instance the data-types "Fuzzy Set" (which is different from the data-type "Set") or "Fuzzy Number" (different from the data-type "Real Number" - some axioms which are true for real numbers are no more valid for fuzzy numbers, see Dubois & Prade 1980), but they are also data types in the ordinary sense. It corresponds to the non fuzzy manipulation of fuzzy quantities.

The author wish to thank Michel CARROT, Henri PARZENY, Louis TERAUD for the role they played in the discussion of the ideas presented here.

A little more fuzzy, is the concept of fuzzy rejection. It corresponds to the ordinary functions equipped with a fuzzy domain A (see Dubois & Prade 1980). It means that all the elements of X are not equally suitable as inputs of the function under consideration. Thus each x is rejected with a degree equal to  $1 - \mu_A(x)$  (if  $\mu_A(x) = 0$  the output of the function is the classical REJECT). We have, thus an example of fuzzy control !

Already intrinsically fuzzy are the data types where some primitive operations or functions are fuzzy. The members are or are not fuzzy. The axioms are classical ones : it is not because an operation is fuzzy that it interacts in a fuzzy manner with itself or others.

Lastly, the axioms of a data type may be themselves fuzzy, even if the operations or/and the members are not fuzzy. What is a fuzzy axiom ? It is a fuzzy description of the way operations are allowed to interact with one another. The description is fuzzy because the allowed interaction is supposed ill-known. For instance, we may say that the operation \* is approximately commutative if  $A * B$  is approximately equal to  $B * A$  for all A and B where "approximately equal" is modelled by a fuzzy relation. \* is approximately commutative in another sense if  $A * B = B * A \vee_{\text{most}}^{\text{for}}$  of the A and B. Fuzzy axioms are linguistic statements on the allowed interactions of the operations. Linguistic statements can be represented using possibility distributions.

Thus there are several kinds of fuzzy data types because, members, operations and even axioms may be fuzzy independently of each other.

Obviously, it is there just some hints. The theory remains to be done.

#### V - CONCLUDING REMARKS

Fuzzy programming does not seem to be either a dream or a dubious speculation. Fuzzy procedures may be useful in many problems of Artificial Intelligence. A language such as LISP seems to be a good support to implement fuzzy procedures after defining some general facilities as fuzzy pattern matching or manipulation of fuzzy numbers.

#### ACKNOWLEDGEMENTS

The author wish to thank Michel CAYROL, Henri FARRENY, Louis FERAUD for the role they played in the discussion of the ideas presented here.

REFERENCES

- . Adamo, J.-M. L.P.L. - A fuzzy programming language.1.Syntactic aspects. Fuzzy Sets and Systems Vol.3 n°2 pp.151-179. 1980
- . Adamo, J.-M. L.P.L. - A fuzzy programming language.2.Semantic aspects. Fuzzy Sets and Systems Vol.3 n°3 pp.261-289. 1980
- . Adamo, J.-M. The L.P.L. language : Some applications to combinatorial and non-deterministic programming. Fuzzy Sets and Systems. To appear.
- . Baldwin, J.F. An automated fuzzy reasoning algorithm. C.N.R.S. Round Table on Fuzzy Sets, Lyon, France, June 23-25, 1980
- . Bonissone, P. A pattern recognition approach to the problem of linguistic approximation in systems analysis. Memo. UCB/ERL M78/57 Berkeley 1978
- . Cayrol, M., Farreny, H., Prade, H. Fuzzy instructions : Execution/Production in "Towards the use of fuzzy set theory in A.I." L.S.I.Tech.Rep.Toulouse. France. Feb.1979
- . Cayrol, M., Farreny, H., Prade, H. Fuzzy reasoning based on multivalent logics in the framework of production-rules systems. Proc.10<sup>th</sup> IEEE Int.Symp.Multivalent Logic. Evanston, Illinois, June 1980. pp.143-148
- . Cayrol, M., Farreny, H., Prade, H. An advanced pattern-matching method taking into account the uncertainty in meaning. Int.Conf.Art.Int. & Information-Control Syst.of Robots, Smolenice near Bratislava. Czechoslovakia.June 30-July 4,1980
- . Chang, C.L. Interpretation and execution of fuzzy programs. In "Fuzzy sets and their applications to cognition and decision processes" (Zadeh,Fu,Tanaka,Shimura eds.) pp.191-218. Academic Press. 1975
- . Chang, S.K. On the execution of fuzzy programs using finite state machines. IEEE Trans.Comput. Vol.21 pp.241-253. 1972.
- . Dubois, D., Prade, H. Fuzzy sets and systems : theory & applications (416 p.) Academic Press. 1980
- . Dubuisson, A. Conception et réalisation du compilateur et de l'exécuteur du langage LPL. Thesis 3rd cycle, Lyon-I, France. Octobre 1979
- . Eshragh, F. Mamdani, E.H. A general approach to linguistic approximation. Int.J.Man-Machine Studies Vol.11 pp.501-519. 1979
- . Fellingner, W.L. Specification for a fuzzy systems modelling language. Ph.D.Thesis, Oregon State Univ. Corvallis. 1978
- . Floyd, R. Nondeterministic algorithms. J.of A.C.M. Vol.14 n°4 pp.636-644.1967
- . Freksa, C. L-FUZZY - an AI language with linguistic modification of patterns. AISB Conference, Amsterdam, July 1980
- . Gershman, A. Fuzzy sets methods for understanding vague hints for maze running. M.S.Th. Comp.Sci.Dept. U.C.L.A. 1976
- . Goguen, J.A. On fuzzy robot planning. In "Fuzzy sets and their applications to cognition and decision processes" (Zadeh,Fu,Tanaka,Shimura eds).Acad.Press 1975
- . Goguen, J.A. Robust programming languages and the principle of maximal meaningfulness. Milwaukee Symp.on Automatic Computation and Control pp.87-90 1976
- . Guttag, J.V., Horowitz, E., Musser, D.R. Abstract data types and software validation. Com.of ACM Vol.21 n°12 pp.1048-1064. 1978

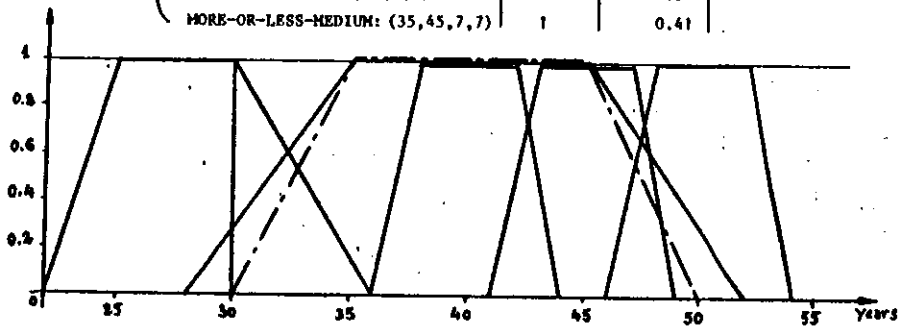
- . Hamilton, M., Zeldin, Z. Higher order software. A methodology for defining softwares. IEEE Trans.Soft.Eng. Vol.2 n°1 pp.9-32. 1976
- . Hinde, C.J. Algorithms embedded in fuzzy sets. Int.Computing Symp. Morlet et Ribbens eds. pp.381-387 North-Holland 1977
- . Jakubowski, R., Kasprzak, A. Application of fuzzy programs to the design of machining technology. Bulletin of the Polish Academy of Sciences Vol.21 pp.17-22 1973
- . Kling, R. Fuzzy-PLANNER : Reasoning with inexact concepts in a procedural problem-solving language. J.of Cybernetics Vol.4 n°2 pp.105-122. 1974
- . Le Faivre, R.A. The representation of fuzzy knowledge. J.of Cybernetics Vol.4 n°2 pp.57-66. 1974
- . Le Faivre, R.A. Fuzzy problem-solving. Ph.D.Th. Univ.Wisconsin. 1974
- . Liskov, B.H., Zilles, S.N. Specifications techniques for data abstractions IEEE Trans.Soft.Eng. Vol.1 n°1 pp.7-19. 1975
- . Mamdani, E.H. F.L.C.S. : A control system for fuzzy logic. Queen Mary Coll. London. 1975
- . Mandic, N.J., Mamdani, E.H. A fuzzy linguistic calculator. 4<sup>th</sup> European Cong. on Operations Research. July 22-25, 1980. Cambridge. England
- . Noguchi, K., Umamo, M., Mizumoto, Tanaka, K. Implementation of fuzzy artificial intelligence language FLOU. Tech.Rep.Autom.Language IECE. Osaka.1976
- . Parnas, D.L., Shore, J.E., Weiss, D. Abstract data types defined as classes of variables. Proc.Conf.on Data. Salt Lake. Sigplan Notices. Vol.8 n°2 pp.149-154. March 1976
- . Prade, H. et al. What 'Fuzzy H.O.S.' may mean.Tech.Rep. Higher Order Software,Inc. Cambridge, MA. 1978
- . Santos, E. Fuzzy and probabilistic programs. In "Fuzzy automata and decision processes" (Gupta,Saridis,Gaines eds.) pp.133-147. North Holland. 1977
- . Shaket, E. Fuzzy semantics for a natural-like language defined over a world of blocks. M.S.Th. Memo n°4 A.I. Comp.Sci.Dept. U.C.L.A. 1976
- . Smith, M.H. et al. The system design of Jason, a computer controlled mobile robot. IEEE S.M.C. Conf. San Francisco Sept.1975
- . Tanaka, K., Mizumoto, M. Fuzzy programs and their execution in "Fuzzy Sets and Their Applications to Cognition and Decision Processes" (L.A.Zadeh, K.S.Fu, K.Tanaka, M.Shimura, eds.) pp.41-76. Academic Press. 1975
- . Umamo, M., Mizumoto, M., Takana, K. FSTDS-system : a fuzzy set manipulation system. Information Science. Vol.14 pp.115-159. 1978
- . Uragami, M., Mizumoto, M., Tanaka, K. Fuzzy robot controls. J.Cyber.6 pp.39-64 1976
- . Zadeh, L.A. Fuzzy algorithms. Information and Control Vol.12 pp.94-102. 1968
- . Zadeh, L.A. Outline of a new approach to the analysis of complex systems and decision processes. IEEE Trans.S.M.C. Vol.2 pp.28-44 1973
- . Zadeh, L.A. PRUF - A meaning representation language for natural languages. Int.J.for Man-Machine Studies Vol.10 pp.395-460 1978

APPENDIX 4 : NUMERICAL DATA OF THE EXAMPLES

1) Universe : age ; tolerance :  $\lambda = 0$

Fuzzy labels :

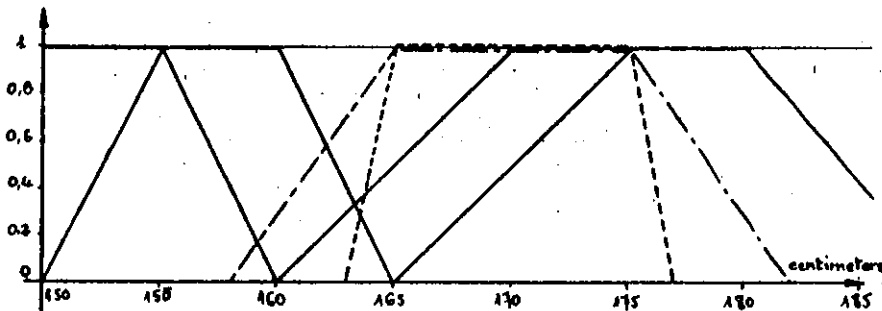
	$\pi(P D)$	$N(P D)$
Pattern : AROUND-FORTY: (35,45,5,5)		
THIRTY: (30,30,0,0)	0	0
ABOUT-FIFTY: (48,52,2,2)	0.57	0
ABOUT-FORTY: (38,42,2,2)	1	1
RATHER-YOUNG: (25,30,3,6)	0.53	0
ABOUT-FORTY-FIVE: (43,47,2,2)	1	0.42
AROUND-FORTY: (35,45,5,5)	1	0.5
MORE-OR-LESS-MEDIUM: (35,45,7,7)	1	0.41



2) Universe : height

Fuzzy labels

	tolerance : $\lambda = 0$		tolerance : $\lambda = 5$	
	$\pi(P D)$	$N(P D)$	$\pi(P D)$	$N(P D)$
Pattern MEDIUM: (165,175,2,2)				
TALL: (175,210,10,15)	1	0	1	0
MEDIUM: (165,175,2,2)	1	0.5	1	0.77
Data RATHER-TALL: (170,180,10,10)	1	0	1	0.11
SMALL: (155,160,5,5)	0.28	0	0.58	0
VERY-SMALL: (150,155,5,5)	0	0	0.15	0



3) Universe : weight

tolerance :  $\lambda = 0$

	$\pi(P D)$	$N(P D)$
Pattern OVER-SEVENTY: (70,120,2,0)		
AROUND-SIXTY: (55,65,3,3)	0	0
ABOUT-FIFTY: (47,53,2,2)	0	0
Data FIFTY: (50,50,0,0)	0	0
ABOUT-SEVENTY-FIVE: (72,78,4,4)	1	0.67
SIXTY-NINE-AND-HALF: (69.5, 69.5, 0,0)	0.75	0.75

