CHOICE OF A SUITABLE ETALON IN DECISION PROCESSES OF DISCRETE SYSTEMS WITH AMBIGUITY

Vladimír OLEJ, Department of Computer Science, Technical University, 031 19 Liptovský Mikuláš, Czechoslovakia

Abstract: The paper brings an example of application programming language fuzzy PROLOG (FPROLOG) in decision processes of discrete systems with ambiguity.

1. Introduction

Knowledge, that is being evaluated by a man, is often characterized by ambiguity. Moreover, this knowledge is often expressed not accurately.

The programming language FPROLOG anables to process clauses (facts and rules) on the basis of generalized truth values. If the individual clauses are confident (they occur with the truth value 0 or 1), then they are processed by the classical programming language PROLOG. While the basis of the programming language is the resolution principle in the predicate logic of the first order [1], the basis of the programming language FPROLOG is the fuzzy resolution principle [2 - 4] in the propositional fuzzy logic.

2. Example of the Choice of a Suitable Etalon by Programming Language FPROLOG

Let two features be given that may have values from 1 to 5, three etalons out of which the first one has feature values 1 and 3, the second 3 and 2 and the third 5 and 1. Let us choose the membership function of features in the following way:

- low feature value	1.0, 0.6, 0.2, 0.0, 0.0
- middle feature value	0.0, 0.4, 1.0, 0.4, 0.0
- high feature value	0.0, 0.0, 0.2, 0.6, 1.0

Then it is necessary to find the most suitable etalon for ambiguous stating of feature values (low, middle, high value). The program in the programming language FPROLOG can be designed as follows:

```
 10 choice('LL',Et):-p1(Et,H1),p2(Et,H2),low(H1),low(H2).
 20 choice('LM',Et):-p1(Et,H1),p2(Et,H2),low(H1),middle(H2).
 30 choice('LH',Et):-p1(Et,H1),p2(Et,H2),low(H1),high(H2).
 40 choice('ML',Et):-p1(Et,H1),p2(Et,H2),middle(H1),low(H2).
 50 choice('MM',Et):-p1(Et,H1),p2(Et,H2),middle(H1),middle(H2).
 60 choice('MH',Et):-p1(Et,H1),p2(Et,H2),middle(H1),high(H2).
 70 choice('HL',Et):-p1(Et,H1),p2(Et,H2),high(H1),low(H2).
 80 choice('HM',Et):-p1(Et,H1),p2(Et,H2),high(H1),middle(H2).
 90 choice('HH',Et):-p1(Et,H1),p2(Et,H2),high(H1),high(H2).
100 p1(Et,H12){1/(1,1),1/(2,3),1/(3,5)}repeat#3.
110 p2(Et,H12){1/(1,3),1/(2,2),1/(3,1)}repeat#3.
120 low(H12){1/(1),0.6/(2),0.2/(3),0/(4),0/(5)}repeat#5.
130 middle(H12){0/(1),0.4/(2),1/(3),0.4/(4),0/(5)}repeat#5.
140 high(H12){0/(1),0/(2),0.2/(3),0.6/(4),1/(5)}repeat#5.
```

In the statements 10 to 90 there are deriving rules for individual combinations of features. Low feature values are denoted L, middle M and high H. In the statements 100 and 110 there are etalons and in the statements 120 to 140 there are membership functions of feature values.

3. Results of the Program

```
*?-choice(Name,Etalon).
(Name = 'LL',Etalon=1){MT=0.2, T=0.0, with CONF=0.6}
(Name = 'LL',Etalon=2){MT=0.4, T=0.0, with CONF=0.2}
(Name = 'LL',Etalon=3){MT=0.0, T=0.0, with CONF=1.0}
(Name = 'LM',Etalon=1){MT=1.0, T=1.0, with CONF=1.0}
(Name = 'LM',Etalon=2){MT=0.4, T=0.0, with CONF=0.2}
(Name = 'LM',Etalon=3){MT=0.0, T=0.0, with CONF=1.0}
(Name = 'LH',Etalon=1){MT=0.2, T=0.0, with CONF=0.6}
(Name = 'LH',Etalon=2){MT=0.2, T=0.0, with CONF=0.6}
(Name = 'LH',Etalon=3){MT=0.0, T=0.0, with CONF=1.0}
```

```
(Name = 'ML' ,Etalon=1){MT=0.2, T=0.0, with CONF=0.6}
(Name = 'ML' ,Etalon=2){MT=0.6, T=1.0, with CONF=0.2}
(Name = 'ML' ,Etalon=3){MT=0.0, T=0.0, with CONF=1.0}
(Name = 'MM' ,Etalon=1){MT=0.0, T=0.0, with CONF=1.0}
(Name = 'MM' ,Etalon=2){MT=0.4, T=0.0, with CONF=0.2}
(Name = 'MM' ,Etalon=3){MT=0.0, T=0.0, with CONF=1.0}
(Name = 'MH' ,Etalon=1){MT=0.2, T=0.0, with CONF=0.6}
(Name = 'MH' ,Etalon=2){MT=0.0, T=0.0, with CONF=1.0}
(Name = 'MH' ,Etalon=3){MT=0.0, T=0.0, with CONF=1.0}
(Name = 'HL' ,Etalon=1){MT=0.2, T=0.0, with CONF=0.6}
(Name = 'HL' ,Etalon=2){MT=0.4, T=0.0, with CONF=0.2}
(Name = 'HL' ,Etalon=3){MT=1.0, T=1.0, with CONF=1.0}
(Name = 'HM' ,Etalon=1){MT=0.0, T=0.0, with CONF=1.0}
(Name = 'HM' ,Etalon=2){MT=0.4, T=0.0, with CONF=0.2}
(Name = 'HM' ,Etalon=3){MT=0.0, T=0.0, with CONF=1.0}
(Name = 'HH' ,Etalon=1){MT=0.2, T=0.0, with CONF=0.6}
(Name = 'HH' ,Etalon=2){MT=0.2, T=0.0, with CONF=0.6}
(Name = 'HH' ,Etalon=3){MT=0.0, T=0.0, with CONF=1.0}
```

From the results of the program it follows that, if there exists an etalon that meets the requirements accurately (cases LM,ML,HL) it will be manifested in the high value of the parameter MT that is called the truth of the consequent [2 - 4]. In further cases (LL,MM,HM) etalon with the closest value can be choosen again by means of the maximal value MT (in our case 0.4). In other cases the maximal values are MT = 0.2 what can be explained by the fact that no one of the given etalons meets ambiguous values of featers to a greater degree. The parameter T gives the truth of the proposition and the parameter CONF is the coeficient confidence [2 - 4].

From the results of the program it can be seen that for etalon number 1, having been determined by feature values 1 and 3, the corresponding ambiguous feature values are low and middle. For etalon number 2, having been determined by feature values 3 and 2, the corresponding ambiguous feature values are middle and low. For etalon number 3, having been determined by feature values 5 and 1, the corresponding ambiguous feature values are high and low. The above results from the program where T = 1.0.

This example can be used in the case, when it is necessary to solve the problem of the choice of a suitable etalon in decision processes of discrete systems on the basis of ambiguous requirements.

4. Conclusion

The program of the choice of a suitable etalon was written in the programming language FPROLOG and it was verified on the microcomputer TNS/AT.

The programming language FPROLOG can be used for designing fuzzy expert systems, fuzzy inferential mechanisms, in robotics, in recognition, in decision process of discrete systems [5] and other fields of artificial intelligence.

References

[1] ROBINSON,J.A.: A Machine Oriented Logic Based on the Resolution Principle. J. ACM, Vol.12, No1, 1965, pp.23-41.

[2] SHEN,Z.-DING,L.-MUKAIDONO,M.: A Theoretical Framework on Fuzzy Prolog Machine. Fuzzy Computing, Theory, Hardware, and Applications, North-Holland, Amsterdam, 1988, pp.89-100.

[3] SHEN,Z.-DING,L.-MUKAIDONO,M.: Demonstration on the Fuzzy Prolog System. Preprints of Second IFSA Congress, Tokyo, 1987, pp.844-847.

[4] MUKAIDONO,M.-SHEN,Z.-DING,L.: Fuzzy Prolog. Preprints of Second IFSA Congress, Tokyo, 1987, pp.452-455.

[5] CHMÚRNY,J.-OLEJ,V.-LEHOTSKÝ,M.: Application of Programming Language Fuzzy Prolog in Decision Processes of Discrete Systems. Elektrotech. Čas., 42, No 5, 1991, (In Press).