Knowledge Base and Inference Motor for an Automated Management System for developing Expert Systems and Fuzzy Classifiers

JESÚS SÁNCHEZ, FRANCKLIN RIVAS, JOSE AGUILAR
Postgrado en Ingeniería de Control y Automatización
Universidad de los Andes
Facultad de Ingeniería
Mérida − Venezuela 5101

Abstract: This article presents the development of the knowledge base and inference motor for an Automated Management System for developing Expert Systems and Fuzzy Classifier (SAGSECD). SAGSECD is a tool that can be used for any user with basics knowledge in the expert systems or fuzzy classifiers area for designing and implementing this kind of systems to solve several problems or situations that require it. This tool works in different platforms and operative systems, and moreover it is supplied with a graphical user interface (GUI) that makes easy the interaction between the designer of the expert or fuzzy system and the tool.

Key−Words: Expert systems, fuzzy logic, knowledge base, inference motor, database, programming language.

1 Introduction

Expert Systems are knowledge−based computer programs designed to resolve problems that require human experts to be solved. A human expert is a person that has deep knowledge about a certain theme (generally specific), and experience solving important problems. Most computers, nowadays, execute a big quantity of programs that realize logical decisions and use a few quantity of knowledge. This programs are divided in two parts: algorithms and data. The algorithms determinate the steps to be followed for solving a certain problem and the data characterize the parameters of a particular problem. The human experts do not follow this model to solve a problem, they use fragments of knowledge and experience to reach the solution of a particular problem. Expert systems represent this fragments of experience and knowledge in a knowledge base, for using it later for solving a particular problem. As a consequence of this, expert systems differ from the conventional programs in their architecture, in the method used to incorporate the knowledge, in the interactive way to be executed and in the impression created in the users that use them; generally these programs responds like an human expert. The expert systems have the capability to solve difficult problems as well or better than a human expert; they heuristically process the information using the rules considered efficient by the experts, interact efficiently and in natural language with the people, manipulate symbolic expressions and reason about them, work with erroneous data and imprecise rules, contemplate multiples alternative hypothesis and explain their conclusions.

2 Problem Formulation

It is presented the design and implantation for an Automated Management System for developing Expert Systems and Fuzzy Classifiers (SAGSECD, Spanish abbreviation), that is a tool that can be used to design and implement expert systems and/or fuzzy classifiers systems for solving problems or situations in any area that require it. This tool must work in different platforms and operative systems.

The system will be divided in five components: rules editor, knowledge base, facts base, inference motor and adaptive system.

The objective is to construct the knowledge base and the inference motor for this system.

The structure that will present the tool its shown below:

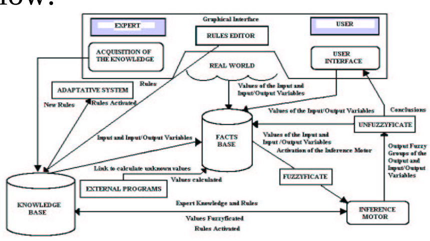

Figure 1 SAGSECD Structure.

3 Problem Solution

3.1 Knowledge Base

The knowledge base is represented by a relational database that contains some parameters required for the system for designing and implementing expert systems and fuzzy classifiers. To introduce this data is used a graphical user interface (GUI) generated using a high level language [2,3,5,11].

In accordance with the kind of system that will be designed, the necessary parameters for the knowledge base differ.

Required parameters for an Expert System:

- Variable name.
- Kind of variable: input, output or both.
- Rank of values, if is necessary.
- Variable engineering units, if is necessary.

Required parameters for a Fuzzy Classifier System:

- Variable name.
- Kind of variable: input, output or both.
- Rank of values.
- Variable engineering units, if is necessary.
- Fuzzy groups names of the variable.
- Membership function for every fuzzy group.

The knowledge is represented in a procedural [1] form that uses production rules, which specify what to do and when.

To accomplish the portability requirements of the system it was used MySQL [4,6] as relational database.

To allow the design and use of several systems on the same computer or work station, it is necessary that every knowledge base generated has a particular name assigned.

The knowledge base has a group of tables that contain the necessary information for implementing an expert or fuzzy classifier system.

These tables differ in accordance with the kind of system that will be designed.

3.1.1 Fuzzy Classifier System Tables

3.1.1.1 Table Input_base

This table keeps the name, rank and units of the input or input/output variables.

Field	Type	Null	Key	Default	Extra
Variable	varchar(25)	YES		NULL	
Rank	varchar(25)	YES		NULL	
Units	varchar(25)	YES		NULL	

Table 1 Description of the Table Input_base

3.1.1.2 Table Input Variable

Each variable of the table Input_base generates automatically other table with the name of the variable; this new table keeps the fuzzy group name, the membership function and the membership level.

Field	Type	Null	Key	Default	Extra
Linguistic_value	varchar(25)	YES		NULL	
Fuzzy_set	varchar(255)	YES		NULL	
Membership_level	double	YES		NULL	

Table 2 Description of the Table Input Variable.

3.1.1.3 Table Output_base

This table keeps the name, rank and units of the output or input/output variables.

Field	Type	Null	Key	Default	Extra
Variable	varchar(25)	YES		NULL	
Rank	varchar(25)	YES		NULL	
Units	varchar(25)	YES		NULL	

Table 3 Description of the Table Output_base.

3.1.1.4 Table Output Variable

Each variable of the table Output_base generates automatically other table with the name of the variable, this new table keeps the fuzzy group name, the membership function and the discrete membership function (group of values that represent the ownership grade of each value of the discreet output variable universe).

Field	Type	Null	Key	Default	Extra
Linguistic_value	varchar(25)	YES		NULL	
Fuzzy_set	varchar(255)	YES		NULL	
Discrete_membership _function	varchar(255)	YES		NULL	

Table 4 Description of Table Output Variable.

3.1.1.5 Table Input/Output Variable

The input/output variable can be found in the tables Input_base and Output_base, and the table generated with the name of this variable keeps the fuzzy group name, the membership function, the ownership grade and the discreet ownership function.

Field	Type	Null	Key	Default	Extra
Linguistic_value	varchar(25)	YES		NULL	
Fuzzy_set	varchar(255)	YES		NULL	
Membership_grade	double	YES		NULL	
Discrete_membership _function	varchar(255)	YES		NULL	

Table 5 Description of the Table Input/Output Variable.

3.1.1.6 Table Users

To offer more security, this table keeps the different kind of users of each system designed.

Field	Type	Null	Key	Default	Extra
Designer	varchar(25)	YES		NULL	
Normal	varchar(25)	YES		NULL	
Special	varchar(25)	YES		NULL	
System	varchar(25)	YES		NULL	

Table 6 Description of the Table Users.

3.1.2 Classic Expert System Tables

In this case, the tables generated with the name of the input, output and input/output variables do not exist, the other tables are the same.

3.2 Inference Motor

For the inference motor, there has been created several functions that emulate a reasoning mechanism able to deduce a solution using a group of input data and information stored in the knowledge base. To accomplish the portability requirements of the system, the inference motor was codified in the programming language Java [2,5,11]. Two inference motors were generated, one to the fuzzy classifier system and the other to the classic expert system.

3.2.1 Fuzzy Classifier System Inference Motor

The Design was based on the Mamdani−type Reasoning Mechanism applied for MIMO Models [1]. It uses the forward chaining [1] and the statistical reasoning [1] to verify the activation grade of each rule. The output variable solution is obtained applying the area centre method [1], to the respective aggregation function. The ranks of values for the output variables have been previously normalised independently of the symmetry or asymmetry that them present [12], and the universes of the output variables were divided in forty one (41) points due to is the maximum number of values supported for the table where they are kept.

3.2.2 Classic Expert System Inference Motor

It uses the forward chaining [1] and monotonic reasoning [1] to verify the activation grade of each rule and to generate the corresponding solution. If during the inference is necessary to make comparisons, operations and/or mathematical functions with the input and/or output variables, the inference motor will solve it. The mathematical operators that can be manipulate are: +, −, *, /; the comparison operators: >, <, >=, <=, !=, ==, and all the mathematical functions supported by MySQL [4,6].

Both inference motors chain rules, they allow the use of input/output variables. When the rules are constructed, each input or output variable can be connected with the others using the operators "AND" ", "OR", "NOT", "AND NOT", "OR NOT".

3.3 Graphical Interface

It facilitates the interaction between the designer of the expert or fuzzy system with the tool. In this interface is introduced the minimal information necessary to create the system to be implemented, moreover the solution that the inference motor offers is presented in a panel that contains the values of the output variables of each activated rule. Depending of the kind of user, is added to the solution another panel that contains the rules activated for the system during an inference process with the respective activation grades. Just like the inference motor, the graphical interface was codified in the programming language Java [2,3,5,11] to accomplish the portability requirements of the system.

3.3.1 Panel CreateUser:

It allows the creation of three kind of users, the revocation of their permissions on a knowledge base, and the access to the system. The three possible kind of users are:

- Designer: has the faculty to design, modify and use the expert or fuzzy classifier system chosen.
- Special User: uses the expert or fuzzy classifier system chosen for receiving a solution and explanation about the rules activated with the respective activation grade.
- Normal User: uses the expert or fuzzy classifier system chosen just for receiving a solution.

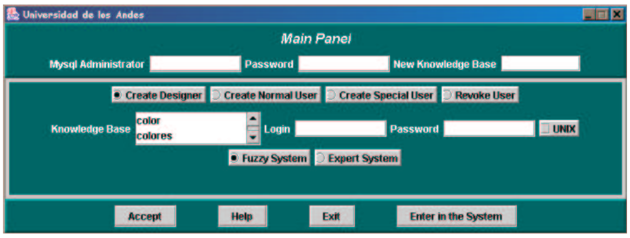

Figure 2 Panel CreateUser.

3.3.2 Panel KnowledgeBase:

This panel is able to design or modify a knowledge base of a system chosen, and allows the link with others tool modules like the facts base, the rules editor and the adaptive system.

For designing and modifying the knowledge bases, the panel offers the following options:

- New knowledge base.
- Show knowledge base.
- Delete knowledge base.
- Add variable.
- Edit variable.
- Delete variable.

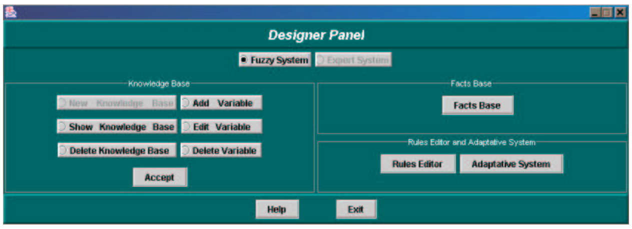

Figure 3 Panel KnowledgeBase.

3.3.3 Panel NewKnowledgeBase:

This panel is activated when a new knowledge base is going to be created or a new variable is going to be added, and it also receives all the information necessary concerning the variables that conform the system.

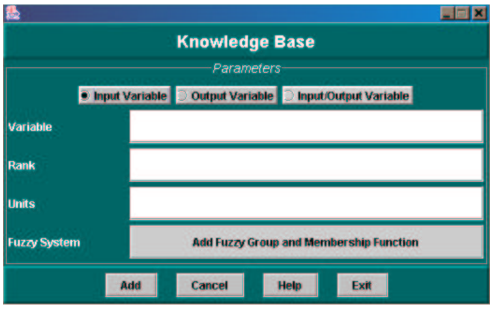

Figure 4 Panel NewKnowledgeBase.

3.3.4 Panel AddLingValueFuzSet:

In this panel the fuzzy groups and the respective membership functions are added to the variables of the panel NewKnowledgeBase or EditKnowledgeBase.

Only the next functions can be added:

- Gaussian Function: membership function defined by the standard deviation and the mean, or by the minimum and maximum values that can take the function.
- Triangular Function: membership function defined by the three values of the abscissas axis that generate the triangle, automatically the values 0,1,0 are set in the ordained axis.
- Trapezoidal Function: membership function defined by the four values of the abscissas axis that generate the trapezoid, automatically the values 0,1,1,0 are set in the ordained axis
- Half Trapezoid Function: membership function defined by the three values of the abscissas axis that generate the initial or final half trapezoid, automatically the values 1,1,0 or 0,1,1 are set in the ordained axis respectively.
- Other Functions: membership function defined by straight lines whose values are introduced in a table.

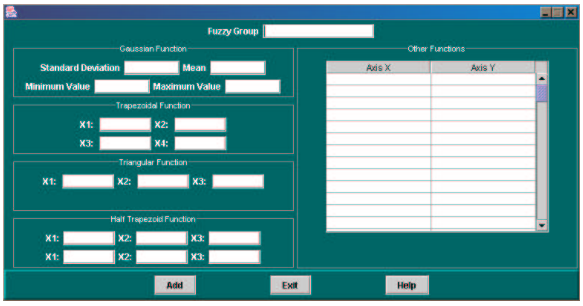

Figure 5 Panel AddLingValueFuzSet

3.3.5 Panel Edit :

This panel is activated to choose a variable and the information about it that will be modified. The next options are offered for editing variables:

- Rename variable.
- Change rank.
- Change units.
- Rename fuzzy group.
- Change membership function.
- Add fuzzy group.

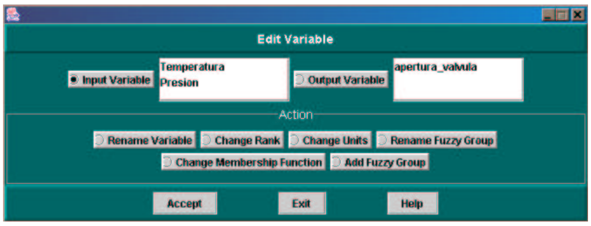

Figure 6 Panel Edit.

3.3.6 Panel EditKnowledgeBase:

The information that will be modified in the variable is introduced in this panel.

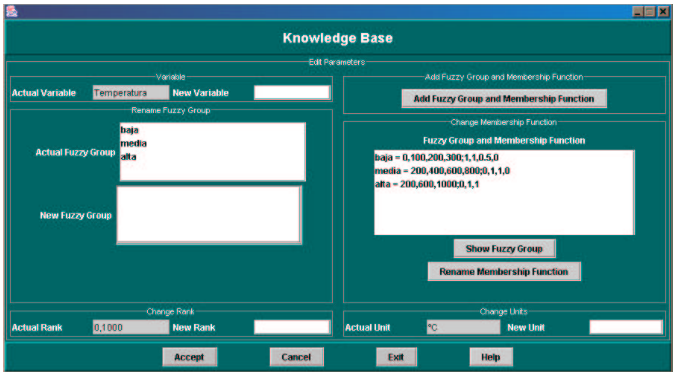

Figure 7 Panel EditKnowledgeBase.

3.3.7 Panel Delete:

This panel is activated to choose a variable or the fuzzy group of a variable to be deleted. It offers the next options:

- Delete Variable.
- Delete fuzzy group.

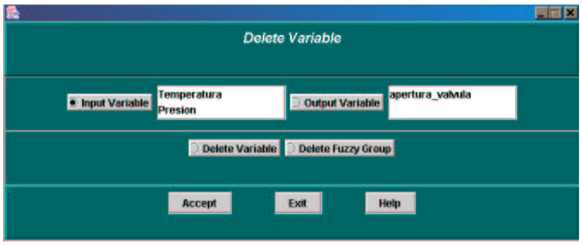

Figure 8 Panel Delete.

3.3.8 Panel DeleteKnowledgeBase:

The information to be deleted of a variable is chosen in this panel.

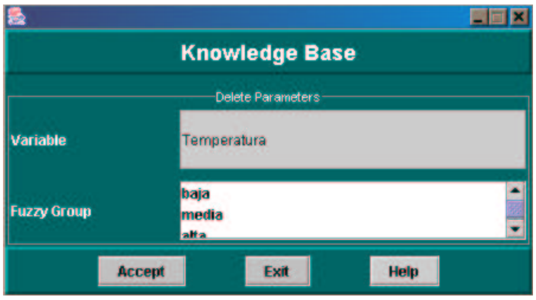

Figure 9 Panel DeleteKnowledgeBase.

3.3.9 Panel Show:

This panel allows to choose all the content of a variable from the knowledge base to be shown to a user.

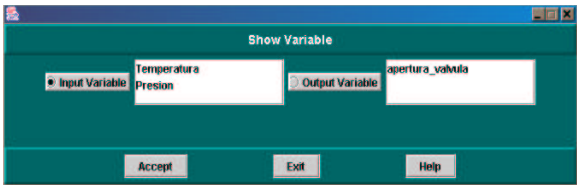

Figure 10 Panel Show.

3.3.10 Panel ShowKnowledgeBase:

This panel shows the rank, the units, the fuzzy groups and membership functions of the chosen variable in the fuzzy system case. In the case of a expert system, this panel shows only the rank and the units.

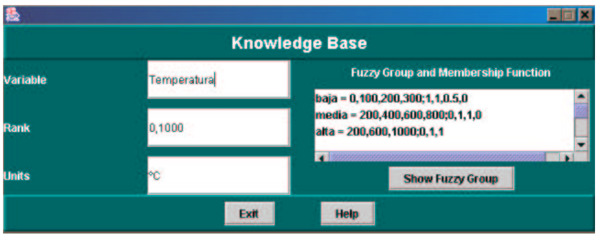

Figure 11 Panel ShowKnowledgeBase

3.3.11 Panel ActivatedRules:

This panel shows the rules activated by the inference motor and their respective activation grade.

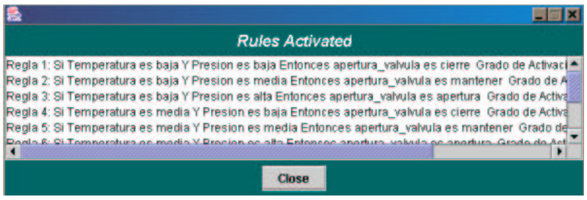

Figure 12 Panel ActivatedRules.

3.3.12 Panel InferenceMotor:

This panel shows the solution obtained by the inference motor for a fuzzy classifier system.

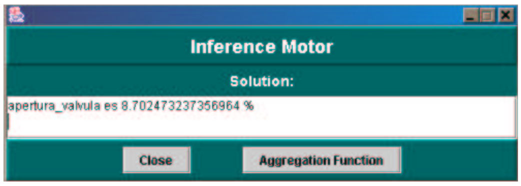

Figure 13 Panel InferenceMotor.

3.3.13 Panel AggregationFunction :

This panel allows to select the aggregation functions of the output variables activated by the inference motor to be shown to the user.

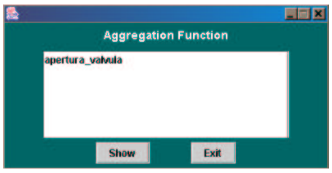

Figure 14 Panel AggregationFunction

3.3.14Panel ShowAggregationFunction:

This panel generates the scheme of the aggregation function chosen.

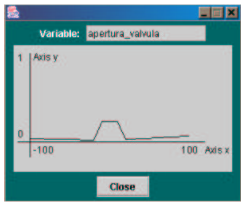

Figure 15 Panel ShowAggregationFunction.

3.3.15 Panel InferfenceMotor2:

This panel shows the solution obtained by the inference motor for a classic expert system.

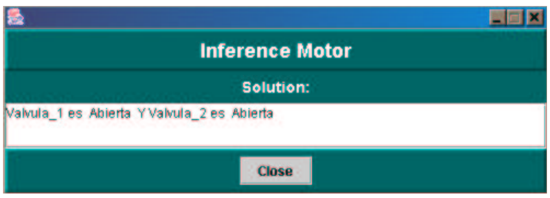

Figure 16 Panel InferenceMotor2.

3.3.16 Panel ShowFuzzySet:

This panel generates the shape of the fuzzy group chosen.

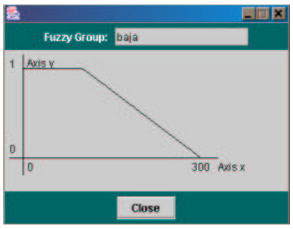

Figure 17 Panel ShowFuzzySet.

To accomplish the portability requirements of the system it was used a driver 100% Java/Protocol Native [3,6] to connect the database with the graphical user interface designed in Java.

3.4 Necessary Packages

− A development environment for Java like Java 2 SDK, which is available in java.sun.com. The standard version of the SDK 1.4 includes the JDBC API [5].

− A MySQL database server that allows the access with a user name and password [6].

− The JDBC controller for MySQL, Connector/J [6].

4 Conclusion

The Automated Management System for developing Expert Systems and Fuzzy Classifiers (http://www.sistemas.ing.ula.ve/~sagsecd), is a tool that will facilitate, to the users with basics knowledge in the area, the design and use of expert systems and fuzzy classifiers.

For accomplishing the latest tendency requirements in software development, this tool presents the characteristics of portability and its development was given using open source software.

References:

[1] Jose Aguilar and Francklin Rivas. *Introducción a las Técnicas de Computación Inteligente*, Editorial MERITEC, Mérida, Venezuela, 2001.

[2] Herbert Schildt, *Manual de referencia Java™ 2*, Osborne McGraw−Hill, Spain, Fourth Edition, 2001.

[3] Phil Hanna, *Manual de referencia JSP*, McGraw−Hill Osborne Media, Spain, 2002.

[4] MySQL AB, *MySQL Reference Manual*, MySQL AB, Sweden, 2003.

[5] "Java Home Page". Sun Microsystems. http://www.java.sun.com

[6] "MySQL Home Page". MySQL AB. http://www.mysql.com/downloads/index.html

[7] Zadeh L. (1965), "Fuzzy Sets", *Information and Control*, vol. 8, pp. 338−353.

[8] Cohen, P. *The Handbook of Artificial Intelligence*. New York. Addison Wesley. 1989.

[9] Harmor, P. Expert Systems: *Artificial Intelligence in Business*. New York. John Wiley & Sons Inc. 1988.

[10] Hayes R. et al. *Building expert systems*. New York. Addison Wesley. 1983.

[11] Mark Watson, *Intelligent Java Applications for the Internet and Intranets*, Morgan Kaufmann Publishers, 2001.

[12] Ronald Yager and Lotfi Zadeh, *An Introduction to Fuzzy Logic Applications in Intelligent Systems*, Kluwer Academic Publishers, 1993.