Qualitative Study of Software for Fuzzy Systems Simulation and Development

EDERSON LUIS POSSELT; LUIS FELIPE DA ROSA E SILVA; REJANE FROZZA; ROLF FREDI MOLZ
Computer Science Department
Universidade de Santa Cruz do Sul – UNISC
Av. Independência, 2293, Santa Cruz do Sul, RS
BRASIL
edersonlp@yahoo.com.br; luisfelipe.silva@terra.com.br; frozza@unisc.br; rolf@unisc.br;
http://www.unisc.br

Abstract: - Fuzzy logic has as main feature the ability to operate on inaccurate data showing the most appropriate ways to deal with approximate reasoning rather than accurate. There are several tools on the market that help businesses and academics in developing fuzzy applications, but they are poorly advertized. This article is an analysis of ten of those tools for fuzzy systems available, pointing out advantages and disadvantages of each one.

Keywords: - Fuzzy logic, fuzzy systems tools

1 Introduction

Fuzzy logic has as main feature the ability to operate on imprecise data [1], being more appropriate to treat approximate modes of reasoning rather than accurate. This technique can return satisfactory results for various applications, and it becomes visible through publications produced on the subject at meetings, completion of course work, thesis, or as reasoning or case study.

Currently, even with the benefits of using fuzzy logic, this technique is not widespread in the software houses as an aid in processes, data which is inaccurate and / or require an update automatically, without human intervention. This is due mostly by the modeling and implementation of a fuzzy system, which requires time, skilled and knowledgeable about the subject.

To improve understanding, learning and application of fuzzy logic, there are available a lot of tools at a market, each with its advantages and disadvantages. With these tools, professionals and enthusiasts of artificial intelligence can create their fuzzy models without the need to invest long hours of programming to obtain the first results of applying the technique in real applications.

This paper aims to present a study of the main available tools, citing the main aspects of each tool and, finally, outline a comparative summary of the most relevant information. As a result of this work, expected to disclose the existing systems and clarify the differences between them.

The article is divided into the following sections: section 2 presents the software studied, in section 3 presents a comparative summary of the studied tools, in Section 4 provides the conclusions of the work.

2 Analyzed Tools

In this paper was analyzed the following tools: FuzzyF, Fuzzy Logic ToolBox, Mathematica Fuzzy Logic, FIDE, TILShell, FuzzyTECH, RockOn Fuzzy Tool, SciLab Fuzzy Fuzzy Tool, UNFUZZY e XFuzzy. The choice of the software mentioned is due to be available for download on the Internet and its uses in projects published [2].

The positive, negative and differential points for these software's are presented in the following sections.

2.1 FuzzyF – Fuzzy Logic Framework

The FuzzyF tool was developed by Master in Computer Science by Joao Ricardo Bittencourt, in *Universidade do Vale do Rio dos Sinos* (UNISINOS). The central idea of this project was to elaborate a set of classes in Java that implements fuzzy logic procedures [3]. The strengths of this project are:

- Free and open source-based GNU GPL license.
- Multiplatform, developed in Java language.
- Extensible code, allowing others to continue the implementation.
- Own documentation for systems integration with the developed classes.
- Samples to be applied and tested, available in the author's website.

The tool has the following weaknesses:

- It has only implemented the model of Mamdani.
- It has only three defuzzification methods (Center of Gravity, Center Point Area and Average Maximum).
- It has no graphical interface for model building.
- It is indicated only for programmers with knowledge of fuzzy logic.
- The definition of the model has to follow the text file format, in a standard defined by the author.
- Last update available for download on June 26, 2006.

2.2 Fuzzy Logic ToolBox

Fuzzy Logic Toolbox is a software module developed by MathWorks MatLab, founded in 1984 in the United States in the city of Natick, Massachusetts. This module offers a complete set of tools for manipulating fuzzy systems, both for model building and for the simulation [4].

The construction of fuzzy models can be accomplished in three ways: the first being through commands, the second through a graphical interface, and the third through block diagrams. The graphical interface provided is very intuitive, encouraging learning.

The system allows export source code in C for embedded software projects. The tool also allows to perform simulations of features, these may be accompanied by visual form. The designer can configure the system so that the values corresponding to the inputs of fuzzy controller are obtained through an external interface.

A strong point of the system is the simplicity in the construction of fuzzy systems. In the project time is possible to define the algorithms and methods to be used in the model. An example of this statement would be the possibility of choosing the defuzzification method, and this setting is clear and simple to be performed. Another example is the possibility of choosing the type of inference, that are being implemented the models of Mamdani and Takagi-Sugeno.

An important fact to remember is the documentation. The user guide is extremely thorough, leaving well-defined the mode of the system's operation, besides having many complete examples and explained. Currently, there are approximately 1,200 books that mention about the MatLab and Simulink tool, which makes clear the potential of this tool.

It is observed as a weak point that the value of software license that currently ranges from around $ 2,500 and the need for computers with high-performance to the system operation. There is the possibility of using versions for students, but these versions have limited functionality. Figure 1 presents a system of screens.

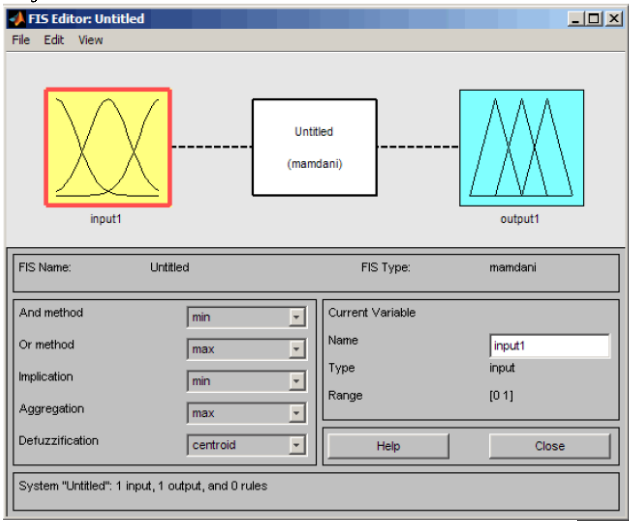

Figure 1 - Fuzzy Logic Toolbox Software Screen
Source: MatLab Manual [4]

2.3 Mathematica Fuzzy Logic

Mathematica is a product developed by Wolfram Research, founded in 1987 by Stephen Wolfram. This is a math software where you can add the module of fuzzy logic [5].

The system developed by Wolfram is complete with respect to features, and implements nearly all, if not all the necessary routines to operate with fuzzy systems.

The Mathematica tool has a way of building fuzzy models only with commands, without graphical interface. This fact makes necessary a greater knowledge by the user to the use of the resources provided by the software.

This system, apart from sin in the absence of a graphical interface for building models, as an aggravating factor has the value of licenses. A license for professional use, is in the range of $ 2,500, without the fuzzy logic package. There are licenses with different values for academic users; while those licenses are acquired by lower values, but this license have a usage time limit. A student license costs in the range of about $ 70 per year.

A strong point of the system is the documentation. The system includes complete manuals with a good description of the features and described samples. Figure 2 shows a screenshot of the software.

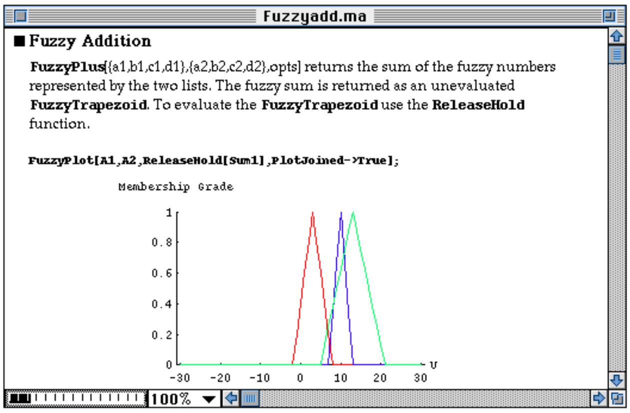

Figure 1 – Mathematica Software Screen
Source: Wolfram Research website[5].

2.4 FIDE

FIDE (Fuzzy Inference Development Environment) is a fuzzy systems tool developed by Aptronix, company founded in Tokyo in Japan in 1988 by a group of mathematicians and engineers. The company has subsidiaries in other countries [6].
This tool has two ways of constructing models, one being through commands and the other through a graphical interface. A differential of this tool is the trace functionality. This feature allows the system to display all operations that are performed during the simulation in order to find and solve problems.
Another interesting detail of this tool is the analyzer functionality, implementation that allows finding errors in the rules in a visual way, and, moreover, indicating the rules that generated them. The system also has the functionality of a simulator, where to use this feature is necessary to create a simply file that serves as a basis for entry into the fuzzy controller. During the simulation process is possible to trace probable errors.
It is possible, through a simple command to generate the source code for the following programming languages: C, Assembly, Java and MATLAB. It can be considered as weakness of this tool, the lack of documentation, compared to other tools like MATLAB and also the fact that there is no free version. Figure 3 shows one of the software screens.

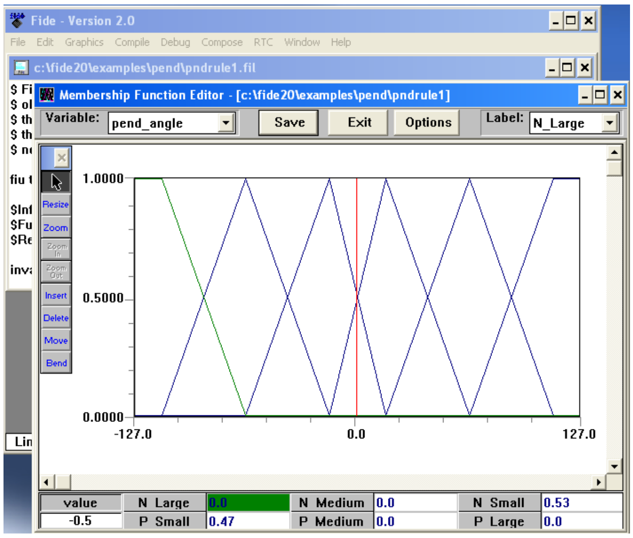

Figure 2 - FIDE Software Screen (Fuzzy Inference Development Environment)
Source: Developer home page [6].

2.5 TILShell 3.0

TILShell is a tool for creating fuzzy models, developed by the Togai InfraLogic, company founded by Dr. Masaki Togai in 1987 [7]. This system has the interface with the user through commands or graphical interface. This system is distributed in three versions, professional, standard and basic. To use the tool you need only a computer with a good configuration.
The system presents interesting features: simulation (function that allows the fuzzy model created can be tested) and generation of source code in C language. This tool is weak in several aspects, they are: Small documentation, which leaves no explicit conditions for acquisition of the tool, without a demo version for download from the internet. In the examples available, it is possible to verify that the system allows the definition of inference algorithms and defuzzification. Figure 4 shows one of the software screens.

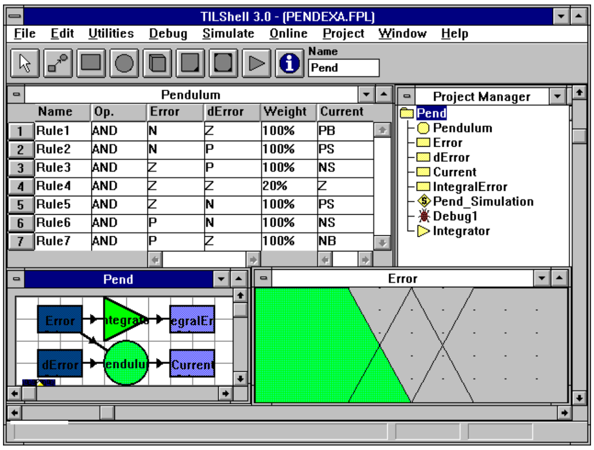

Figure 3 – TILShell Software Screen
Source: Developer's website [7]

2.6 FuzzyTECH

FuzzyTECH is a modeling tool of fuzzy systems developed by INFORM GmbH. An item that differentiates the system from other FuzzyTECH are the characteristics of ergonomics and usability [8].
The creation of the fuzzy controllers is simple, since the creation of the variables and their degrees of relevance to the output variables and defuzzification methods. Based on the input and output variables, the system make easier the creation of rules through a grid previously completed. The system provides tools for analysis and simulation with 2D and 3D graphics resources, besides, allow peripherals integration using RS232 serial port or TCP / IP connections.
Another difference of this tool is the fact that generates documentation on the models of fuzzy systems implemented. The system also has a complete user's manual.
The system is distributed in several versions, which each have certain features, allowing the customer to choose the version that best fits your reality. There is a demo version available on site, but this version has limitations on usage, and one of them is the fact of not allowing the generation of source code in the following languages: C and Java.
You can see as a disadvantage, the fact that this tool is not free, which is can be not viable their use by some users and even educational institutions.
In Figure 5 you can view one of the software screenshots. One of the most interesting features in this model is the easy way to create the fuzzy models. Most transactions occur with drag and drop functions (drag & drop).

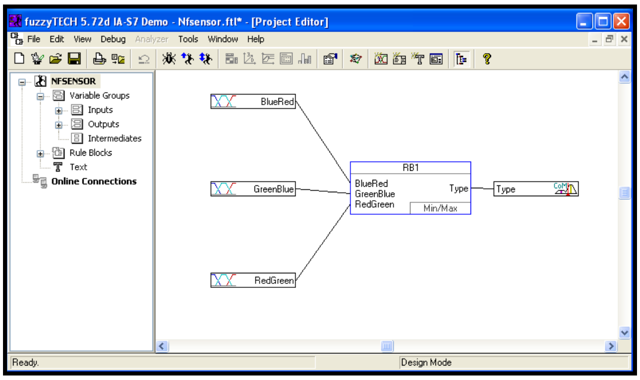

Figure 4 – FuzzyTECH Software Screen
Source: Developer's website [8].

2.7 RockOn Fuzzy Tool

RockOn is an open source software that implements features of fuzzy logic. This software was developed at the University of Kassel in Germany, during a fuzzy logic course, by the Florian Bachmann programmers, Stefan Stützer Stafani and Lind [9].
The software has as main characteristic the project tree on the right side of the screen. This facilitates the creation of fuzzy models and their maintenance. The system makes the practice of Mamdani model and implements the main methods of defuzzification, being: Center of gravity, the average of maximum, Highest of maximum and Lowest of maximum.
The negative points of this tool are: lack of documentation, problems on executing the simulation module, export to a programming language. The positive points are: good graphical interface, multiplatform system and open source. Figure 6 shows one of the software screenshots.

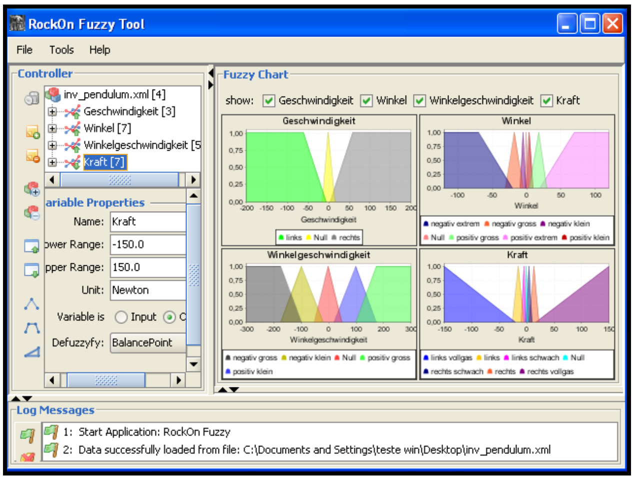

Figure 5 – RockOn Software Screen
Source: Developer`s website [9].

2.8 SciLab Fuzzy Tool

Scilab is a mathematical software that is freely distributed on the Internet and open source since 1994. It is currently a consortium headed by SciLab founded in May 2003 [10].

For the tool can make use of resources of fuzzy logic, is required to download plugins. These increases in SciLab allow the user of the system makes use of fuzzy routines. There are implemented the inference models of Mamdani and Takagi-Sugeno.

We can mention as advantages of this tool the fact that it is free and the possibility of adding new extensions. It means that a programmer can develop their own routines and distribute them on the internet to other system users.

On the downside, is the fact that the extension has no graphical interface available; this demands the user to have a good knowledge of fuzzy logic to initiate activities with this tool. Another weakness is the necessity for the user to have good computer skills to install it.

In Figure 7 you can see the image of one of the screens of the system.

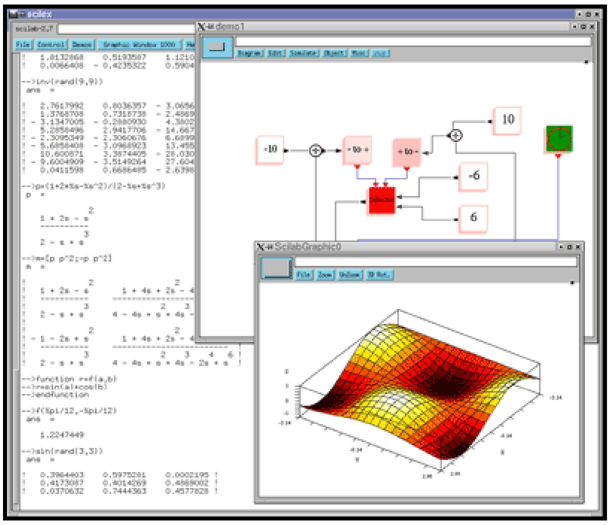

Figure 6 – SciLab Software Screen.
Source: Developer home page [10].

2.9 UNFUZZY – Fuzzy logic system

UNFYZZY (Fuzzy logic system) was developed by Teacher Oscar Germán Duarte Velasco and oriented by Gustavo Perez as master thesis in 1997 by University of Colombia [11].

This software is free distributed and available in four languages: English, Spanish, Italian and Czech. Until this moment, in this paper, only this software has the characteristic of being available in more than one language.

The system is graphic, and has good parameterization of the features of methods. To develop a fuzzy controller using this tool, click on the main components and adjust them as necessary for the application. The implementation of the tool presented some graphical problems such as ghosting.

The hardware requirements to run this software are low. This tool was developed and approved to run on the Windows operating system.

As a disadvantage of this tool can mention the fact that only implement the inference engine of Mamdani. Another disadvantage is the fact that mount fuzzy models of only one inference system. In order to exemplify this part, with the tool FuzzyTech, the output of a fuzzy system serves as the entrance to a next fuzzy system.

We can mention as advantages of this tool, the fact that it exported source code for C and C ++ languages. In addition, the tool includes support for simulation and debugging of fuzzy controllers implemented through the graphical interface available.

Figure 8 displays screens of the software developed by Velasco. There you can see on top a fuzzy controller in full screen and below a parameterization of the inputs of the system.

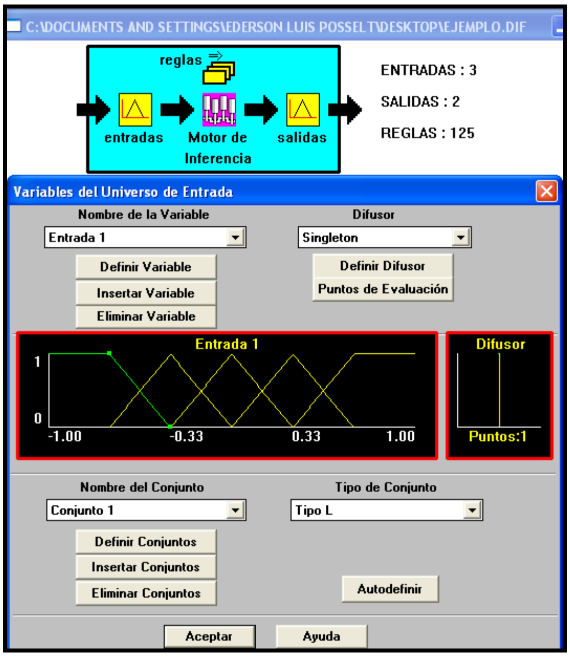

Figure 7 – UNFUZZY Fuzzy logic system developed by Velasco in 1997.
Source: Developer home page [11]

2.10 XFuzzy

XFuzzy is a free tool. The distribution and modifications are controlled by GNU License (General Public License). This tool belongs to the

developers and the IMSE-CNM (Institute of Microelectronic of Sevilla – National Center of Microelectronic) [12].
The system is developed in Java language, which makes it a multiplatform system.
The Figure 9 presents the tool organization.
- Description stage: For the definition of the fuzzy system.
- Verification Stage: Tools for simulation and monitoring.
- Tuning Stage: Learning.
- Synthesis stage: Functionality of export source code to another languages.

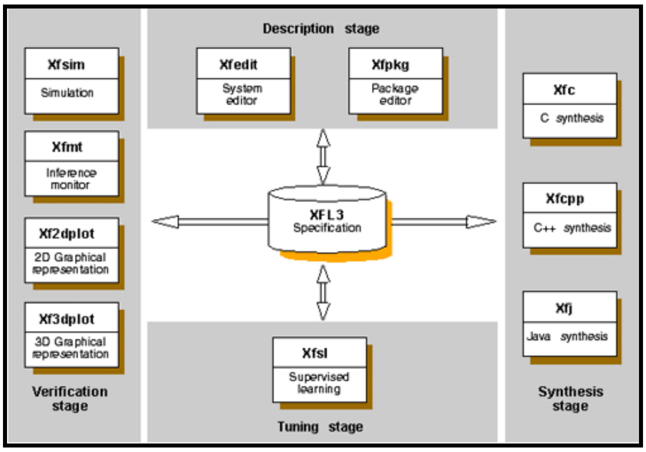

Figure 8 - Organization of Xfuzzy tool
Source: Developer home page [12]

This tool has several strengths.
About the most favorable aspects, can be mentioned:
- Full documentation. There are several videos on the www.youtube.com making demonstration of the tool.
- Free Software and open source.
- Multiplatform. It can be executed on any operating system with Java Virtual Machine (JRE) installed.
- Input rules. Rules can be created in three ways: free (user-created), in table format where the user only fills gaps, and the way junction where the user determines the output for two input variables.
- Export source code in C, C + + and Java.
- Implements the inference models of Mamdani and Takagi-Sugeno.
- Implements 10 defuzzification methods. The Figure 10 presents the initial screen of the Xfuzzy software.

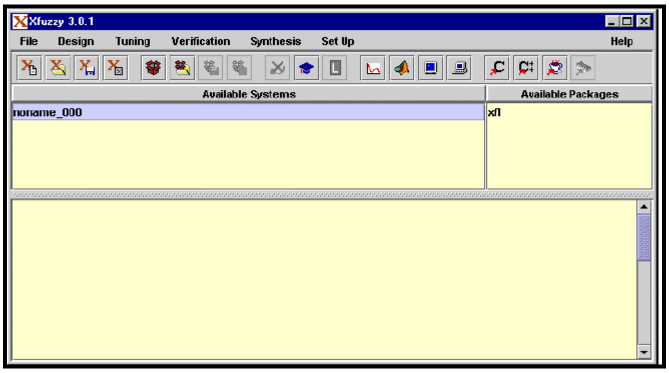

Figure 9 – Principal Screen of Xfuzzy software.
Source: Manual of Xfuzzy software [12]

3 Comparative about the analyzed tools

After analysis of several tools we can come up with some definitions. The first is that there are several systems already deployed in the market, some paid and others free. In the next section will explore the main items that differentiate the best tools in the analysis in this paper.
About the paid tools, the MatLab and FuzzyTech had the largest number of resources and suitable graphic interface. The FuzzyTech tool takes advantage over MatLab in the issue about the export of source code and simulation, where you can obtain external inputs such as serial port.
Among the free tools, XFuzzy proved to be the best. This tool has the features found only on paid software like MatLab and FuzzyTech, and also has open source code, which allows new implementations, which were controlled by the GNU license.
But none of the tools have all the features implemented. Below are listed some important resources distributed on the tools:
Only FuzzyTech has implemented project report routine. This report is automatically generated during the assembly process of the fuzzy controller. This feature is interesting for control and system maintenance.
The MatLab do not have any export to source code for development languages like C and Java. XFuzzy do not generate reports of projects implemented and also offers features not input data for simulation through interfaces such as sockets and serial ports.
Table 2 presents a comparison of the main features of the tools of fuzzy systems analyzed. To be able to view the Table 2, Table 1 provides an index of tools.
It is important to remember that much of the information was drawn from the documentation tools, and it should be because these were not available in demo versions.

Logically, some of the tools have features beyond those listed in Table 2, but the fundamental characteristics were used for the development of fuzzy systems.

Sumary	Software
1	FuzzyF
2	Fuzzy Logic ToolBox
3	Mathematica Fuzzy Logic
4	FIDE
5	TILShell
6	FuzzyTECH
7	RockOn Fuzzy Tool
8	SciLab Fuzzy Fuzzy Tool
9	UNFUZZY
10	XFuzzy

Table 1-Sumary of analyzed tools

	1	2	3	4	5	6	7	8	9	10
Inference machine										
Mamdani	x	x	x	x			x		x	x
Takagi-Sugeno		x	x							x
Self						x				
Fuzzy										
Gaussian	x	x	x			x		x	x	x
Triangular	x	x	x	x	x	x	x	x	x	x
Trapezoidal	x	x	x	x	x		x	x	x	x
Cauchy		x	x					x	x	x
Sigmoid		x	x					x	x	x
Interface										
Graphics		x		x		x	x		x	x
Command Line		x	x	x				x		
Defuzzification										
Center of gravity	x	x	x			x	x	x	x	x
Area Central Point	x	x	x							
Average of Maximus	x	x	x			x	x	x	x	x
First of maximus		x	x				x	x	x	x
Last of maximus		x	x				x	x	x	x
Height		x	x				x	x	x	x
Exports Source Code										
C		x		x	x	x			x	x
JAVA				x		x				x
Assembly				x		x				
License										
Paid		x	x	x	x	x				
Free	x						x	x	x	x
Open source	x						x	x		x
Simulation										
Have simulation		x	x	x	x	x	x			x
Documentation										
Good		x	x			x		x		x
Regular	x			x	x				x	
Poor							x			

Table 2- Comparison of the main features of a fuzzy tool.

4 Final Thoughts

After reviewing the tools presented in this paper, we reach the goal proposed to present and compare some development fuzzy tools. This paper facilitates the work of professionals and students, once they are presented and compared the systems of construction of fuzzy models.

In this article, were not approached items about operation (use mode). To validate items about operation, we need to build fuzzy models in each tool and validate the processing of input, inference and output. This stage is open for future work.

Even with several tools presented, the market lacks a free tool directed to the college, taking into account items of usability and didactic teaching in the implementation of fuzzy models.

References

[1] PATYRA, M. J. et al. Fuzzy Logic: Implementation and Applications. Toronto: Wiley Teubner Publishers, 1996.

[2] USIDA, W. F.. Controle fuzzy para a melhoria do perfil de tensão em sistema de distribuição de energia elétrica. Dissertação de mestrado – Escola de engenharia de São Carlos, da Universidade de São Paulo. São Carlos, 2007

[3] BITTENCOURT, J. R. ; OSÓRIO, F. S.. *FuzzyF - Fuzzy Logic Framework: Uma Solução Software Livre para o Desenvolvimento, Ensino e Pesquisa de Aplicações de Inteligência Artificial Multiplataforma*. In: III Workshop sobre Software Livre, 2002, Porto Alegre. Anais do III Workshop sobre Software Livre. Porto Alegre : Sociedade Brasileira de Computação, 2002. p. 58-61

[4] MATHWORKS. *Fuzzy Logic ToolBox, User's Guide*. MathWorks, 2009. Available at: <www.mathworks.com>. Accessed on, Aug 2009.

[5] WOLFRAM. *MATHEMATICA FUZZY LOGIC, The most flexible environment for exploring fuzzy systems.* WolfFram Research, 2003. Available at: <www.wolfram.com>. Accessed on, Aug 2009.

[6] ZHU, D. *FIDE -Fuzzy Inference Development Environment An Intelligent Software Tool for Mutli-Sensor Data Fusion.* Aptronix Application Note, 1997.

[7] TOGAI INFRALOGIC . *The World's Source For Fuzzy Logic Solutions*. Available at: <http://www.ortech-engr.com/fuzzy/togai.html>. Accessed on, Nov 2009.

[8] FUZZYTECH. *Fuzzy Logic Advantage in Your Design*. Available at: < www.fuzzytech.com>. Accessed on, Nov 2009.

[9] ROCKON FUZZY TOOL, *The Best Free Fuzzy Control And Simulation Tool.* Available at: < http://www.timtomtam.de/rockonfuzzy/>. Accessed on, Nov 2009.

[10] SCILAB, *The open source platform for numerical computation.* Available at: < http://www.scilab.org/. Accessed on, Nov 2009.

[11] VELASCO, O. G. D.. *Sistema de lógica difusa.* Available at: www.ing.unal.edu.co/~ogduarte /softwareDetallado.htm. Accessed on, Dez 2009

[12] IMSE-CNM. *FUZZY LOGIC DESIGN TOOLS.* Software manual. 2003