USING FUZZY LOGIC TO IMPLEMENT ADAPTABILITY IN WAY-Z39.50

Camino Fernández **Vicente Matellán** **Ignacio Aedo** **Paloma Díaz**
Departamento de Informática
Universidad Carlos III de Madrid
Av. Universidad 30, E-28911 Leganés (Spain)
E-mail: {camino@inf,vmo@gsyc.inf,aedo@ia,pdp@inf}.uc3m.es

Summary

The design and implementation of adaptive systems for accessing information is a sophisticated task. Different models have been proposed to build systems that adapt to the user reducing the complexity in the information browsing process, saving user time and offering personalized interfaces. The work described in this paper, named WAY-Z39.50, is an example of fuzzy adaptive system specially built for accessing digital libraries.

Keywords: User interface, Agent, Adaptive system, Z39.50, Digital libraries, Fuzzy application, WAY.

1 INTRODUCTION

Finding and accessing information are complex problems, due to the inherent difficulty about locating appropriate information servers and the information itself; and to the complexity about presenting the information to the user through an adequate interface [1, 5].

To help in the design of this kind of systems, WAY [2] proposes a general model of architecture whose main goal is to improve information access systems through the fulfillment of three tasks: adaptation of the interface to the user, help in the searching process, and extraction and maintenance of updated data about the available information servers.

One implementation of the WAY general model is WAY-Z39.50. It is designed to access servers supporting the Z39.50 information retrieval protocol standard [4], that is being installed in most libraries all over the world to provide access to their catalogues. WAY-Z39.50 is based on an agent-decomposition method where each task of the system is managed by a different agent. Figure 1 shows the five agents that comprise the system, representing by means of dashed ellipses those agents that are instantiated for each user, and by solid ellipses those that are instantiated only once for the whole system.

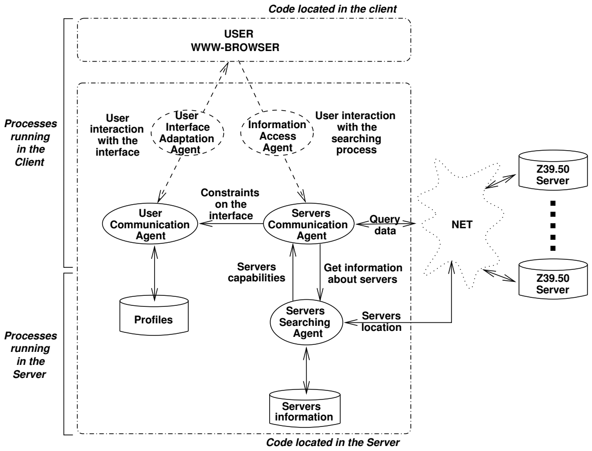

Code located in the client
USER
WWW-BROWSER
Processes running in the Client
User interaction with the interface
User Interface Adaptation Agent
Information Access Agent
User interaction with the searching process
Constraints on the interface
User Communication Agent
Servers Communication Agent
Query data
NET
Z39.50 Server
Z39.50 Server
Servers capabilities
Get information about servers
Profiles
Servers Searching Agent
Servers location
Processes running in the Server
Servers information
Code located in the Server

Figure 1: WAY-Z39.50 architecture

Three of theses agents implement a fuzzy [7] adaptive process (the first three ones in the following description), while the other two ones use crisp processing algorithms:

User Interface Adaptation Agent: The task of this agent is to spy the user actions to decide the best way to adapt the interface to the user. The agent takes decisions and transmits its suggestions to the user communication agent in order to perform them (building the interface). The user can always decide not to allow the agent to change anything.

Information Access Agent: The task of this agent is to spy the user searching process. When a user makes a query to one information server, this agent looks for the servers she prefers and launches parallel searching processes to the ones considered more appropriate by the agent.

Servers Searching Agent: The task of this agent is to maintain updated information about the accessible servers. To perform this task, it can use sources such as web pages, mailing lists or USENET news groups. The agent periodically consults these sources and updates the servers' list. The period is also updated by the agent itself.

User Communication Agent: It is in charge of building the user interface according to user preferences, as well as transmitting the user's actions to the user interface agent following the user interface adaptation agent suggestions.

Servers Communication Agent: It communicates with the Z39.50 information servers, translating user inputs into adequate queries and passing the information retrieved to the user communication agent.

2 WAY-Z39.50 design and implementation

One of the requirements for WAY-Z39.50 implementation was that the software being developed should be broadly available. In order to fulfill this requirement, Java programming language was chosen to make WAY-Z39.50 available through a web page. A java applet implements the interface of the application that is built by the user interface adaptation agent. This agent communicates with the information access agent and the servers searching agent using Java-RMI (Remote Method Invocation) technology to support distribution.

The general model does not force concrete systems to use one particular mechanism to implement the different kinds of adaptability. In WAY-Z39.50, the mechanism for deciding which modifications in the user interface are going to be suggested is based on fuzzy decision systems. Suggestions about user interface comprise both the layout of the components, and their appearance. Besides, fuzzy decision systems will also be used to help the user in the information searching process, as well as in the classification of the Z39.50 servers available.

Fuzzy Relational Algorithms [6] (FRAs) are the method used to store the knowledge required to take the decisions (for instance, which server is going to be used). FRAs are composed by a finite set of fuzzy conditional statements (antecedent and consequent). Antecedent of conditional statements are conjunctions of fuzzy variables defined over inputs. For instance:

IF *Time_Server_1* **is** *Long* **and** *Errors_Server_1* **is** *High* **THEN** *Accessibility_Server_1* **is** *Bad*

The operation named *fuzzyfication* transforms a non-fuzzy quantity into a fuzzy variable. This operation is made using the definition of those variables, that in WAY-Z39.50 has been done using trapeziums [8]. For instance, a value as $Time\_server\_1 = 10msec$ will be translated into activation levels of fuzzy labels as $Fast$, $Slow$, etc.

An extension of the *modus ponens* [3] is used as the inference mechanism to assign values to the fuzzy output variables. These values can be translated into non-fuzzy quantities using a *defuzzyfication* process. The adopted defuzzyfication operation is a version of the "Center of Gravity" procedure [8].

2.1 Designing the fuzzy relational algorithms

Several automatic methods such as neural networks or genetic algorithms have been proposed to generate FRA in different domains. The heuristic method has been the one used in the design of WAY-Z39.50. "Heuristic" means that a human designer will use her experience in order to define the fuzzy subsets (trapeziums in the current implementation) and to write the rules. The main reason for using this approach was to take advantage of the designers' experience.

Three different FRAs have been designed in the current implementation. Each one is used as the processing mechanism of each of the main WAY-Z39.50 agents.

First, the User Interface Adaptation Agent uses a FRA to adapt the user interface taking as inputs the detailed actions of the user during the sessions. The input of this FRA is any action performed by the user on the interface, that is, using one particular searching field, visiting first one kind of records or visiting only the records written in one language. Its output also affects interface aspects as, for instance, placing the searching fields in a different way, ordering the records in a special manner or showing only the records written in the user mother tongue.

Second, the Information Access Agent uses the FRA to decide how many simultaneous searches in different servers are going to be made to speed up the searching process. The agent has to take into consideration, as inputs of this decision system, issues such as the interest of the user for previously proposed searches in that server, the current load of the machine or the accessibility of the alternative servers.

Third, Servers Searching Agent uses a FRA to keep an ordered list of available Z39.50 servers taking into account aspects such as the speed of the communication with that server, its accessibility -as some servers are not available the whole day- or the quality of the results obtained by the user from that server.

2.2 Implementation issues

The software developed is implemented in Java 1.1.5 and it is structured into a hierarchy of packages. The root of this hierarchy is the `way` package, whose main elements are the subpackages `way.interfaz`, that implements the interface building facilities, `way.z3950`, that implements the client side of the Z39.50 protocol, and `way.agents`, that implements the agents of the system. The most relevant java subpackage, that concerns fuzzy issues in WAY-Z39.50 code, is `way.agents.reasoning` which implements the FRA. This implementation of FRA comprises the definition of fuzzy input variables, by means of their name and their membership functions (defined as the four points of a trapezium of height 1) defined in their range. The same format is used to describe fuzzy output variables and their labels.

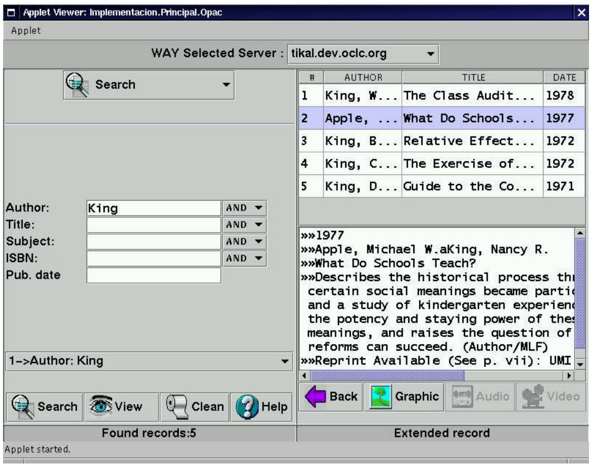
Fuzzy `rules` are also represented as a separate java class made up by the `antecedent` and the `consecuent`. Each of them is composed by a set of `conditions` (represented by another class). These conditions correspond to membership relations. The `rule` class includes a method to implement evaluation which invokes the fuzzyfication of the components, then performs the modus ponens, and finally invokes the defuzzyfication process.

Given this implementation, the definition of FRA consists of the enumeration of the inputs, describing the labels using four points which define the trapezium; the enumeration of the outputs in the same way; and the specification of the fuzzy rules which describe how the outputs are modified depending on the inputs.

3 A session with WAY-Z39.50

When a user first enters the system, she is asked to enter a user name and a password. These data do not have a security purpose, since the system may be freely accessed, but they are only used to identify the user and the agents that are assigned to her (a user adaptation agent and an information access agent). These agents are in charge of the adaptation process and the storage of the user preferences that are used in the reasoning process.

At this point, the user interface adaptation agent builds a particular interface for that user. If this is the first time the user enters the system, a default interface is shown. This first interface depends on the kind of machine the user is connecting from and on the default language selected in the navigator used to access the system. During the session, the user interface adaptation agent is also active and spying user actions. The observation process generates information that is used as input for the fuzzy systems. For instance, the fuzzy reasoning process performed by the user interface adaptation agent can obtain through the fuzzy process a higher value for the `importance` variable of the field `title` than for the field `author`. If the field `author` is displayed in the first position and the field `title` in the second one, this agent would suggest to interchange their positions. Some of the inputs that affect the `importance` of a field are, for instance, the `use` of the field and its `success_rate`. The `use` of the field is the percentage of times that the field is used to make a query and the `success_rate` depends on variables such as `searching_success` that again depends on other variables as `response_time` or `long_records_visited`.



Applet Viewer: Implementacion.Principal.Opac

WAY Selected Server : tikal.dev.oclc.org

Author: King
Title:
Subject:
ISBN:
Pub. date

1->Author: King

#	AUTHOR	TITLE	DATE
1	King, W...	The Class Audit...	1978
2	Apple, ...	What Do Schools...	1977
3	King, B...	Relative Effect...	1972
4	King, C...	The Exercise of...	1972
5	King, D...	Guide to the Co...	1971

```
»»1977
»»Apple, Michael W.aKing, Nancy R.
»»What Do Schools Teach?
»»Describes the historical process thı
 certain social meanings became partic
 and a study of kindergarten experienc
 the potency and staying power of thes
 meanings, and raises the question of
 reforms can succeed. (Author/MLF)
»»Reprint Available (See p. vii): UMI
```

Found records:5

Extended record

Applet started.

Figure 2: WAY-Z39.50 user interface

The user can always accept or refuse proposed changes, as some users do not want computers to take decisions that affect their work without their permission. Let's show a couple of examples about the possible adaptation. A user may have visited only books records so, as the `books_records_frequency` would have a very high value, the agent would propose the user to order the records beginning with the ones corresponding to books. If the same user consults mainly records in Spanish, the `spanish_records_frequency` would get a higher value than the English or French ones, so the agent would propose to order the records beginning with books in Spanish and following with books in English and so on. Figure 2 shows a WAY-Z39-50 customized interface where the user has made a query and is visiting the extended records.

When the user makes a bibliographical query to the system, the first result obtained is the number of records satisfying the query. The user can consult a summary of the results as short records (upper right area of figure 2) meaning that she finds acceptable the number of records obtained. If the information shown in these records seems interesting, she can visit their extended version (bottom right area of figure 2). In this case, it can be supposed that she has found some of the information required or, at least, an interesting part of it, so variables as `server_usage` are increased.

While the user is querying the system, the information access agent is spying the process, which means that it is getting the values of the input variables to the fuzzy reasoning process. This way, the agent has information about the servers that the user visits more frequently or give her back the best results -where "best results" depends on the `server_delay` and `long_records_visited`. While the request is being sent to the Z39.50 server by our system, this agent is deciding and launching new searches on the servers considered more suitable for that user. Among others, variables as `server_accesses`, the number of times that the user has accessed a server, `server_success`, the percentage of successful connections to that server, `server_speed`, the speed of the connection, `long_records_visited`, the number of long records looked up in that server, etc. serve as input variables from which the `server_qualification` is obtained.

The searches launched by the agent are suggested to the user indicating the number of records found matching her query in other servers. This way, the user saves the time that she would have employed in making another query in the same server or going to another server and repeating the query, as it should be done in the classical systems that integrate many library servers.

Finally, the user can choose to go deeper into one of the proposed searches, refine the query or choose another server, beginning the whole process again. From any of these decisions, the agents will collect information that will be used as inputs for future operation as input variables to the fuzzy reasoning process that affect, among others, the `server_qualification`.

4 Conclusions

One of the main advantages of using WAY based systems, and particularly the advantages provided by WAY-Z39.50, is that their interface adapts to the user actions while she uses the system. To do that, the only information source are the user actions themselves, which means that there is no need to ask the user in any moment. They also offer alternative searches in the servers preferred by the user that respond best to information requests, saving user time.

The use of fuzzy logic in the adaptive process has made easier the task of specifying the rules to modify the user interface and to participate in the searching process. The logic rules are written in almost natural language, and the variables and labels involved in those rules have also got a well-known meaning, which makes them easy to design. These rules are also easily accessible as they can be stored on a text file or on a data base. A debugging process makes sure that the rules are correct and notifies the user if they are not. Finally, the reasoning process is really lighter than other mechanisms that require heavy computations, such as genetic algorithms or neural networks.

References

[1] D. Benyon and P. Palanque (1996). Critical Issues in User Interface System Engineering. *Springer-Verlag.*

[2] C. Fernández and P. Díaz and I. Aedo (1998). WAY: An Architecture for User Adapted Access to Z39.50 Servers based on Intelligent Agents. *Proceedings of the European Conference on Digital Libraries 98.* pp. 665-666.

[3] E. H. Mamdani (1984). An analysis of formal logics as inference mechanism in expert systems. *International Journal of Man-Machine Studies.* Vol. 21, n. 1, pp. 213-227.

[4] J. Michael and M. Hinnebusch (1995). *From A to Z39.50: A Networking Primer.* Mecklermedia.

[5] B. Shneiderman (1998). *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* Addison-Wesley Publishing Company, 3rd Edition.

[6] L. A. Zadeh (1973). Outline of a New Approach to the Analysis of Complex Systems and Decision-Making Processes. *Transactions on Systems, Man and Cybernetics.* Vol. 3, n. 1, pp 28-45.

[7] L. A. Zadeh (1988). Fuzzy Logic. *IEEE Computer.* Vol. 21, n. 4, pp. 83-93.

[8] Hans-Jurgen Zimmermann (1990). *Fuzzy Sets. Theory and its Application.* Kluwer Academic Publishers.