

AFM 1.0

USER MANUAL
2nd EDITION

1996

Table of Contents

Before you begin	1
General Conventions	1
Mouse Conventions	1
Keyboard Conventions	1
Introduction to Adaptive Fuzzy Modeller	3
File	4
Learning	5
Build Rule	5
Build Membership Functions	6
Tools	7
Local Rules	7
Simulation	7
Exporter	7
W.A.R.P.-SDT	7
View	8
Directory	8
Window	8
Adaptive Fuzzy Modeller Basics	9
System Requirements	9
Installation	9
Starting to use	10
Choosing Commands from Menus	10
Working in a Dialog Box	11
Using Help	12
File Commands	13
Creating a New Project	14
Project Setting	14
Training and Reporting Files	15
Variables Definition	16
Variables and Fuzzy Sets Renaming	17
Opening an Existing Project	18
Saving the Project	19
Project Information	19
Printing Project Data Structures	20
Printer Setup	21
Exit from AFM	21

Learning Commands	23
Build Rules	23
Using Neural Net	23
Manual Rules Definition	25
Build Membership Functions	26
Mbfs Shape and Inference Method	26
Mbfs Learning	27
View Commands	29
Displaying Rules	29
Displaying Membership Functions	30
Linguistic Format Visualization	30
Graphical Visualization	31
Tools Commands	33
Simulation	33
Simulation on a single pattern	33
Simulation from File	35
Local Rules	37
Visualizing a Rule	37
Inserting a New rule	38
Deleting a Rule	39
Updating a Rule	39
Exporter	41
W.A.R.P.-SDT Exporter	41
W.A.R.P.2.0 Exporter	41
MATLAB [®] Exporter	43
How to use MATLAB [®] Exporter	43
How to use Exporter with MATLAB [®]	43
How to use Exporter with SIMULINK	43
C Language Exporter	44
How to use the C EXPORTER	44
Directory Management	47
Selecting Working Directories	47
Window Commands	49

Appendix A	
Project Files	51
Files Generated by AFM	51
Private Files	51
Reporting Files	52
Learning & Testing Files	53
Exporting Files	54
Appendix B - AFM Tutorial	
Creation of a Fuzzy Model	55
Project Definition	55
Step 1 Project Name and Target processor	56
Step 2 Choose Project Parameters	56
Step 3: Define Variables Parameters	57
Step 4: Variable and Fuzzy Sets Names	58
Learning	59
Build Rules	59
Build Membership Functions	61
Model Simulation	63
Project Exporting	65
Exporting to MATLAB®	65
Exporting to W.A.R.P.2.0	67

Before you begin

Before you start using Adaptive Fuzzy Modeller (A.F.M.), it is important to understand the terms and notational conventions used in this documentation.

General Conventions

The word "Choose" is used to carry out a menu command or a command button in a dialog box.

The chosen commands are given with the menu name preceding the command name. For example, the instance "Choose FILE OPEN" tells you to choose the OPEN command from the FILE menu. This naming convention describes the sequence you have to follow in choosing the command - at first you select the menu, and then you choose the command.

Bold type in text indicates words or characters you type.

Italic type indicates important terms introduced in the section.

UPPER CASE type in text indicates the names of commands and menu buttons of AFM

A numbered list (1, 2, ...) indicates a procedure with two or more sequential steps.

A bullet symbol indicates a list of elements belonging to the same object (for example the fields of a dialog box).

Mouse Conventions

AFM requires one mouse button. The default one is the left button but you can change the setting to the right button with the Control Panel. For information on changing the mouse button, see your operating system documentation.

"POINT" means to position the mouse pointer until the tip of the pointer rests on what you want to point to on the screen.

"CLICK" means to press and immediately release the mouse button without moving the mouse.

"DOUBLE CLICK" means to press the mouse button twice in rapid succession.

"DRAG" means to press the mouse button and hold it down while you move the mouse.

Keyboard Conventions

A plus sign (+) used between two keys indicates that you must press both keys at the same time. For example ALT+F means that you press the ALT key and hold it down while you press the F key; this is the shortcut to choose the File menu.

A comma (,) between two keys indicates that you must press those keys sequentially. For example, "Press ALT, F, O" means that you press the ALT key and release it, press the F and release it, and then press the O and release it. This is the shortcut to choose the FILE OPEN command.

Introduction to Adaptive Fuzzy Modeller

Adaptive Fuzzy Modeller (AFM) is a software tool for the automatic fuzzy model generation of a system starting from its Input-Output data sampling.

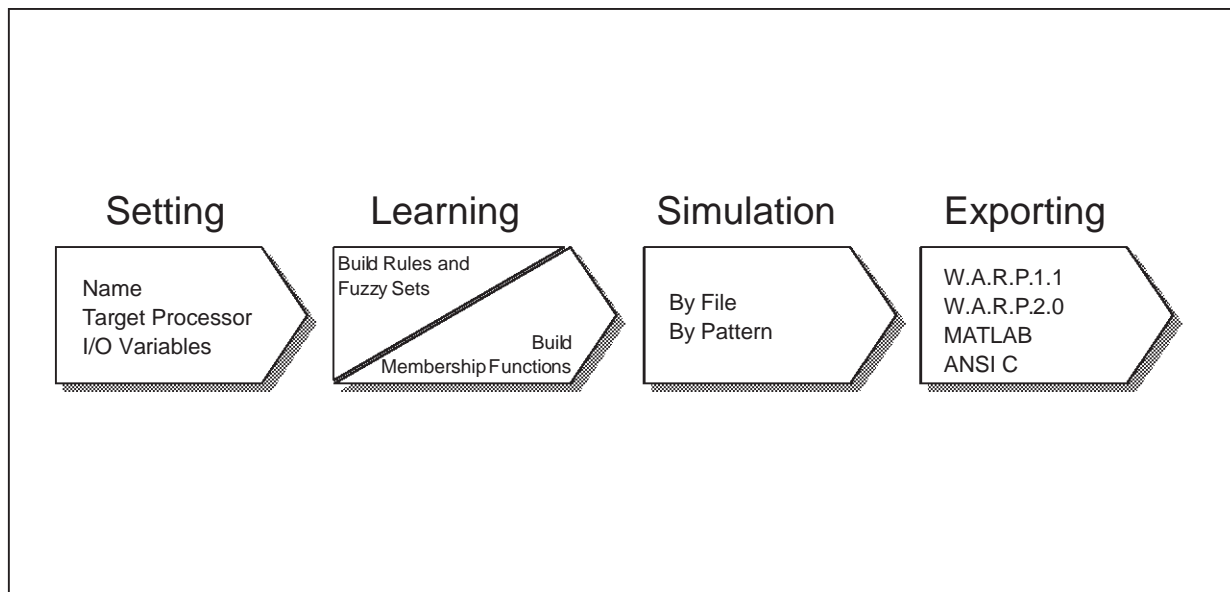
The AFM design flow consists of the following main steps:

Project setting Definition of project name, target processor, number of Input/Output variables and number of related fuzzy sets.

Learning Building of the rules and tuning of each Membership Functions position and shape associated with the fuzzy sets.

Simulation Computation of the consequents related to the given Inputs in order to verify the validity of the project.

Exporting Project translation to W.A.R.P.1.1 and W.A.R.P.2.0. development systems, MATLAB and C code, in order to use the related environment as a support for simulation and control of the so modelled process.



File

It is the first menu to be used in order to start working on a project. It allows you to define the process parameters and to supply the process samples file for the fuzzy model design.

Selecting NEW, the fuzzy system configuration can be initialized. If the project has already been defined, the option OPEN must be used to access to the related file. Once the project has been selected, the other menus are activated and you can go on working. The command SAVE and SAVE AS allow to release the allocated resources and to terminate the current project elaboration saving the project information. The command PROJECT INFO supplies you all the information concerning the project while the PRINT facilities allow you to print the project data. Selecting the command EXIT the AFM will be closed and the control will pass to Windows Environment.



Learning

The Learning phase consists of two different steps: Build Rule and Build Membership Functions.

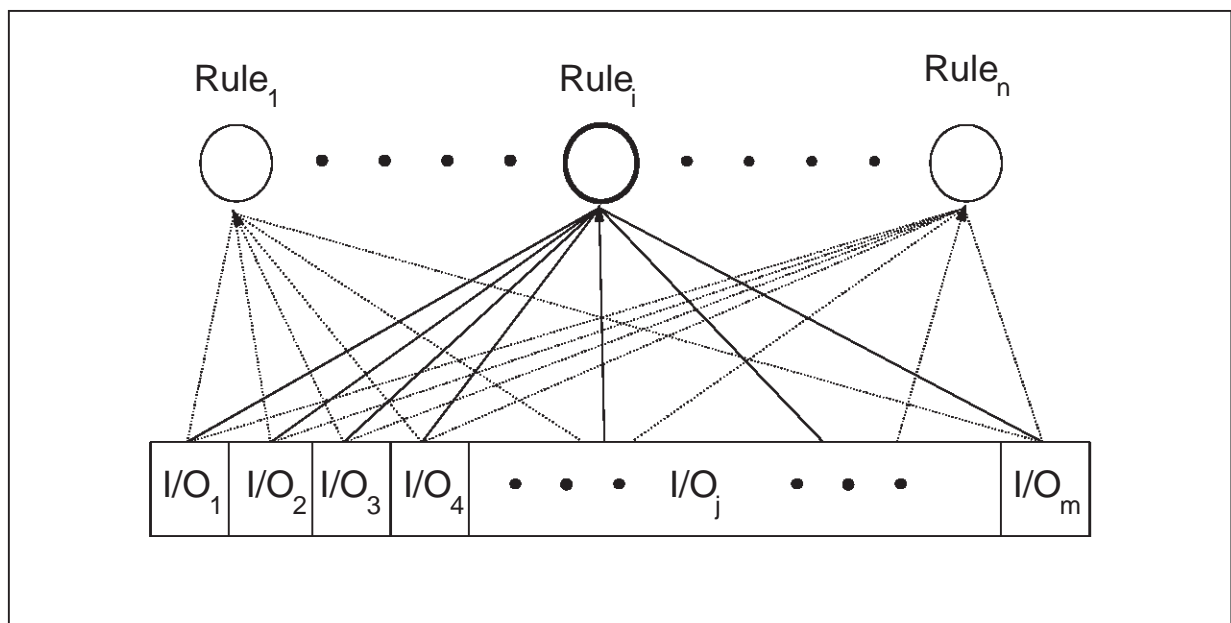
Note that it is necessary to carry out the building of the rules before building the Membership Functions (Mbfs), because the parameters needed for the last phase are generated during the former phase.

Build Rule

The automatic fuzzy rules identification is carried out thanks to an unsupervised learning on a Winner-Take-All Fuzzy Associative Memory (WTA_FAM) neural network. Moreover, this step initializes the Mbfs position that will be optimized during the next step.

Otherwise, if you are an expert on your system, you can reduce the learning cycle rate by initializing manually the fuzzy sets and rule parameters.

As a result you will be supplied with a rule file containing the linguistic expression of the rules.

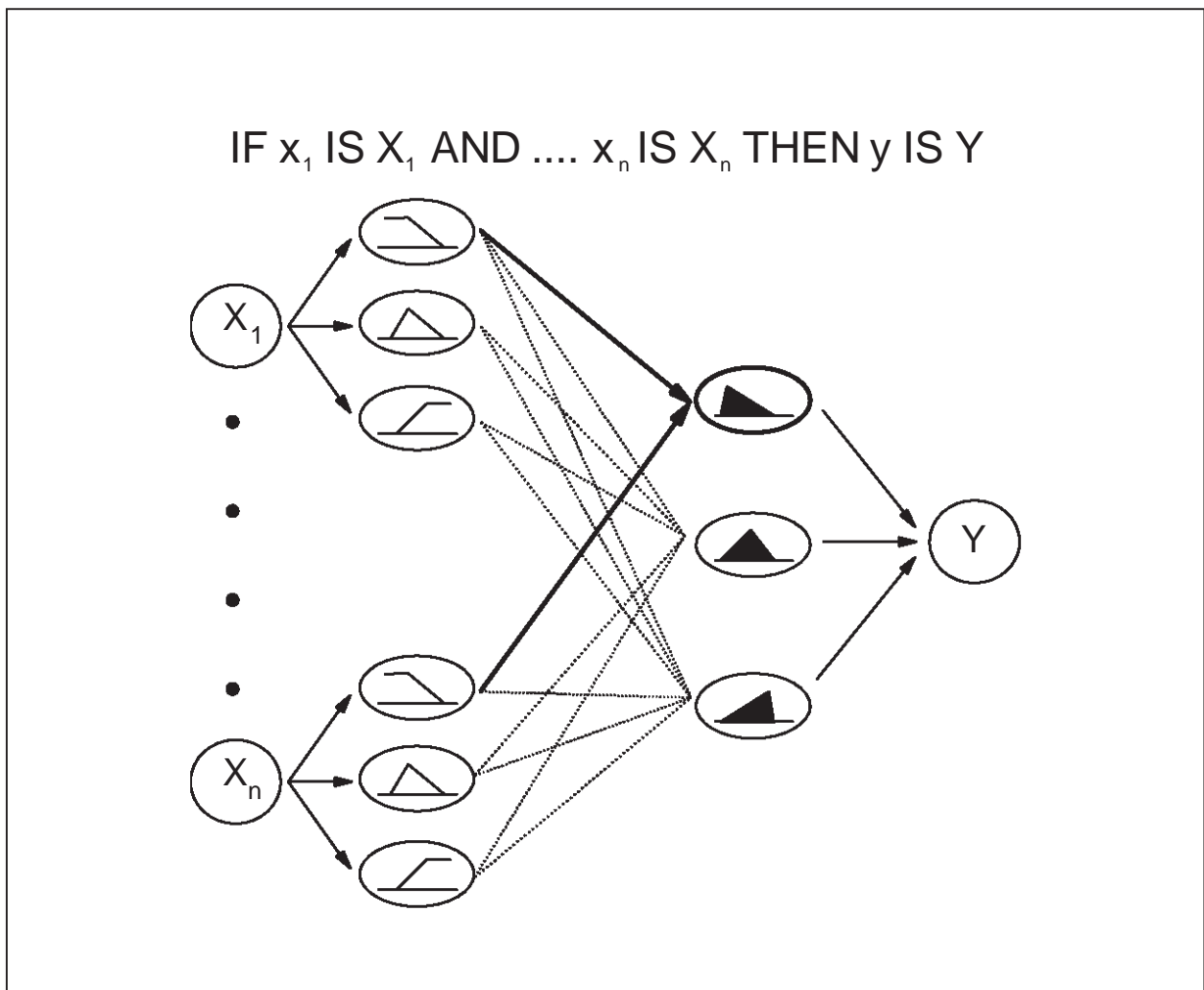


Build Membership Functions

Allows to select the Mbf shape and the fuzzy intersection method for the project elaboration. Starting from the rule file supplied by the previous phase (BUILD RULE), this command initially associates to each fuzzy set a standard Mbf shape which will be tuned during the learning phase in order to let the fuzzy system better approximate the process/function samples by means of subsequently running sessions.

The Mbfs setting is carried out by a Supervised Learning on a multilayer Backward-propagation Fuzzy Associative Memory (BACK_FAM) neural network that identifies the position and the shape of each Mbf.

At the end of this phase you can graphically see the complete fuzzy set associated to each project variable.



Tools

It is composed by different sub-menus:

Local Rules

Allows you to add local rules in order to tune your project in particular system condition. This could be pointed out as an error because of the imposed hardware limitation

Simulation

Allows you to simulate the fuzzy system behaviour in order to verify the approximation level obtained during the learning phase.

The simulation can be carried out in two different ways:

Simulation on a single pattern:

You must supply the simulator with the values of the variables corresponding to the point to verify;

Simulation from file:

You must supply the simulator with the file name of a pattern containing an input test signal and the relative output that will be used to perform a complete process inference.

Exporter

Provides a coding of the project to be imported in other programming environments in order to carry out its simulation and validation.

Supplied options are the MATLAB, C, FU.L.L. and W.A.R.P.-Software Development Tool conversions. Once you have selected an option the new project model is automatically generated by the Exporter.

W.A.R.P.-SDT

The W.A.R.P.-SDT RUN command allows to start the W.A.R.P.1.1 Software Development Tool directly from the AFM window, in order to edit or compile your project after it has been translated by the related exporter.

View

It provides a visualization of the fuzzy model extracted during the learning phase. The MbF of each fuzzy set is described through their analytical expression or through their graphical representation. The inference rules are described through their linguistic expression.

Directory

Provides an easy way to define the path for each file involved in the project. Used during the elaboration of a project, it allows you to state the directories that the dialog box will access directly while opening or saving the project files, avoiding you to move among the directories in order to choose the files location.

Window

Provides an easy way to manage with the graphical windows related to the current application. The provided facilities are: Tile, Cascade, Close All and Minimize All. The Tile and Cascade commands allow to dispose all the windows one next to the other or one upon the other. The Close All discards all the graphical windows and the Minimize All reduces each of them to icons.

Adaptive Fuzzy Modeller Basics

Before using Adaptive Fuzzy Modeller, you first need to learn some basic skills. This chapter gives instructions for installing and starting AFM, using the online Help, the mouse and the keyboard to choose the commands and to select the dialog box options.

System Requirements

Before you install AFM, make sure you have all the hardware and software you need to run the program:

- Intel type 80386 processor or higher (80486 recommended)
- 2 Mbytes RAM memory
- Hard Disk or Floppy disk with at least 1 Mbyte of free space
- VGA or higher graphics card
- Mouse
- Ms-DOS version 3.3 or higher
- Windows version 3.1 or later

Installation

Your first step in using AFM is to use the Setup program to install AFM on your hard disk. Be sure to start Microsoft Windows before installing the AFM.

- 1 Insert the installation disk (labelled "Setup") into the floppy disk drive A or B.
- 2 Once Windows is running, select the Run command from the PROGRAM MANAGER FILE menu.
- 3 Type **A:\SETUP** in the Command Line box then choose OK or press ENTER
- 4 A Window mask will appear, containing a window with the "Initializing setup" message. If you do not want to continue the installation, choose the EXIT button, else choose OK.
- 5 A dialog box appears to set the directory where you want to install AFM. If a directory is not specified, the directory C:\AFM will be automatically generated.
- 6 After copying the necessary files, the setup program automatically generates the group and the icon to start the program.

When the setup procedure is completed, you go back to the Program Manager.

Starting to use

After installing AFM, you can start your design project.

To Start AFM:

- 1 Double-click AFM group icon in the Program Manager window, or open the group that contains AFM.
- 2 Double-click AFM icon to start it.

Choosing Commands from Menus

AFM commands are grouped in menus. The menu names appear in the menu bar displayed on the top of AFM window.

To choose a command with the mouse or the keyboard, first select the menu and then choose the command you want. The following figure shows AFM windows with the File menu selected.

Each underlined character in menu and command names corresponds to the key you can press to select a menu or to choose a command.

In AFM, commands are referred to by the menu and the command name. For example the OPEN command on the File menu is called the File OPEN command.

Choosing a command by using the mouse

If you use the mouse you have to follow this steps to choose a command:

- 1 To display a menu containing the command you want, click the menu name in the menu bar.
- 2 Click the command name.
You can also point to the menu, drag to the command you want, and then release the mouse button.
- 3 If the command displays a dialog box, specify the information you need, (see "Working in a Dialog Box").
- 4 When you finish with the dialog box, click the appropriate button to carry out the command.

Choosing a command by using the keyboard

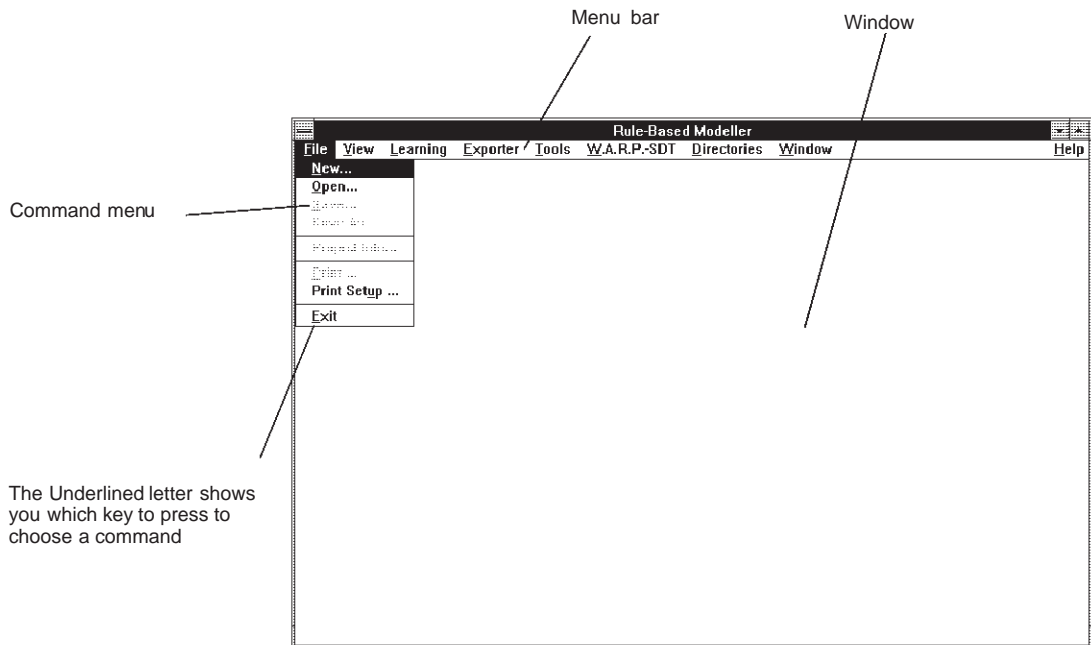
If you want to choose a command with the keyboard:

- 1 Press the ALT key to activate the menu bar.
The File menu appears selected, indicating that the menu bar is active.
- 2 To open a menu press the key of the underlined letter of the command name you want to select or use the left or right arrows keys to select the menu name and the down arrow key to open it.
- 3 If the command displays a dialog box, specify the appropriate information you need. Notice that you can change field by pressing the TAB key.

To close a menu, click anywhere outside the menu and the menu bar or press the ESC key.

To cancel a command, click the CANCEL button on the dialog box or press the ESC key.

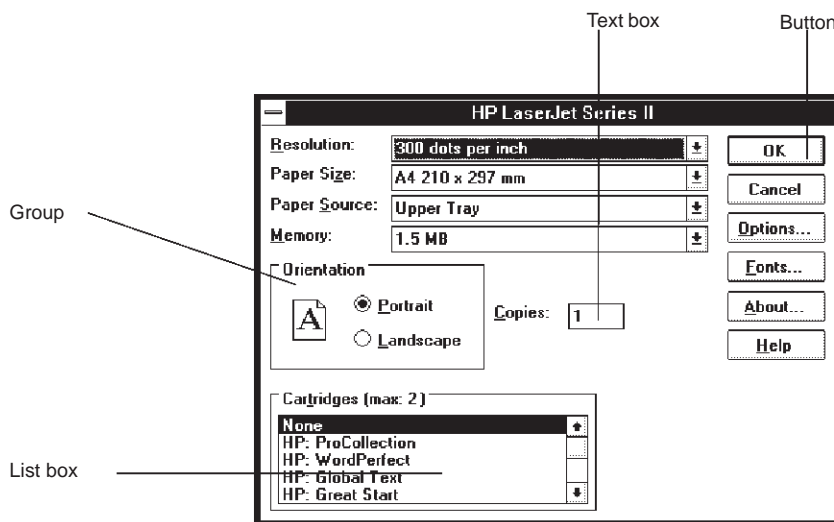
A command name followed by the symbol ... on a menu indicates that AFM displays a dialog box.



Working in a Dialog Box

When you choose a command having options, AFM shows a Dialog Box. A Dialog Box may include areas in which you enter text or numbers and view or change settings for option related to the command.

Some dialog box options occasionally appear dimmed. This indicates that the option is not currently available.



Using Help

AFM is provided with a complete online reference tool. Help is especially useful when you need information quickly or when your User Guide is not available. Help contains information about each command and dialog box.

When you are working with AFM, you can select HELP by double clicking the help item on the menu bar or choosing the HELP button that appears on the dialog box.

Once you are in Help, there are different ways you can move to other topics to find exactly the information you want, including:

Jump Terms. Jump terms are underlined with a solid line and are used to go to the topics in the HELP window.

Back button. The Back button is used to step back through all the topics you have viewed since opening the HELP window.

Index button. The Index button is used to display the list of Help Topics.

Search button. By clicking the Search button a dialog box appears allowing you to select and view the Topic you want help with.

Browse buttons. The Browse buttons allow to forward and backward logical movement among the Topics. The buttons are respectively labelled >> and <<, and the movement follows a logical predefined order.

File Commands

AFM allows project files to be managed by means of the FILE menu. The management of these files is performed according to the standards typical of Windows programs.

Choose the FILE menu to enter file management operations, and select the following items from the sub-menu:

- NEW: closes the actual project and opens a new project.
- OPEN: loads an already defined project from disk.
- SAVE: saves into the disk the actual running project.
- SAVE AS: saves into the disk the actual running project while allowing destination drive, directory and name to be defined.
- PROJECT INFO: supplies all actual project file information.
- PRINT: allows to print the fuzzy data structures of the current project.
- PRINT SETUP: allows to select and configure the printer.
- EXIT: closes AFM application.



Creating a New Project

To create a new project choose FILE NEW command. Everything previously defined will be discarded from memory. If the project has been modified after the last saving, after selecting NEW a save request message box will appear. Choosing NO the project will be discarded without being saved, choosing YES it will be saved according to the information defined in the save request dialog box allowing you to go on with the definition of a new project.

Project Setting

Choosing OPEN command a dialog box will appear asking you to define the main parameters of your project.

The NEW PROJECT dialog box contains:

- the PROJECT NAME edit box
- the TARGET list box

BROWSE tool helps you to define the path and the name of your project files.

Select from the TARGET list box the processor on which you are stating the project. This will affect the dimension of the problem you are dealing with, because of the hardware limitation imposed by the target processor chosen.

Up to now the following targets are available:

GENERAL PROCESSOR

max. 8 antecedents, max. 4 consequents, max. 2^{14} rules.
Gaussian, Scalene and Isoscele Triangular Mbfs.
Inference Method: Minimum or Product.

W.A.R.P. 1.1

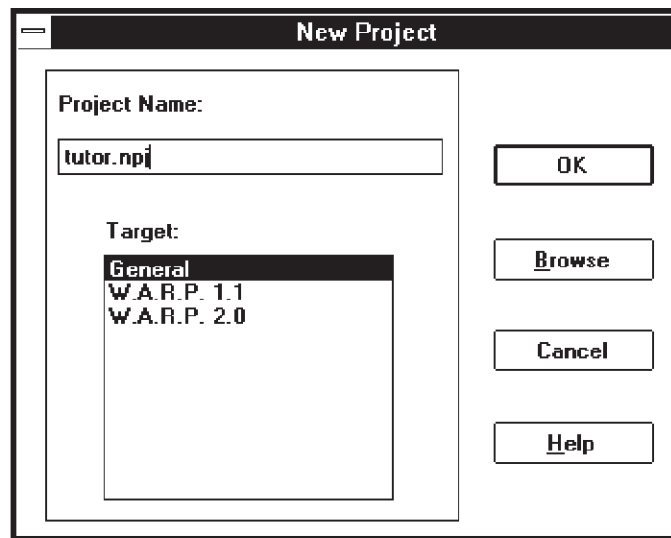
max. 4 antecedents, max. 2 consequents, max. 256 rules.
Gaussian, Scalene and Isoscele Triangular Mbfs.
Inference Method: Minimum.

W.A.R.P. 2.0

max. 8 antecedents, max. 4 consequents, max. 256 rules
Scalene and Isoscele Triangular Mbfs.
Inference Method: Minimum.

In case of a wrong name or of problems during the operation, an error message will appear.

Once stated the choice, click the OK button to carry on with the initialization of the project or CANCEL to discard FILE NEW procedure.



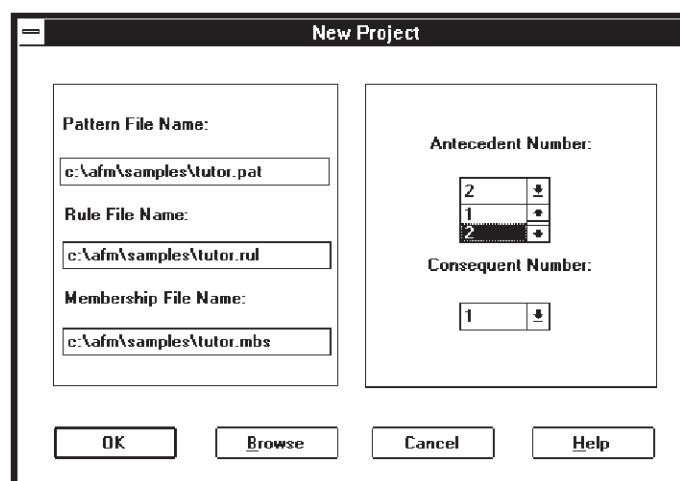
Training and Reporting Files

The generation of the project requires a training set of patterns that link input values to the output ones. You specify the training file - characterized by the extension .PAT - in the PATTERN FILE box text.

AFM generates two reporting files, in ASCII format, containing all data about rules and variables in order to help you in project understanding and checking.

These files are stored by default in the project directory by using the extensions .RUL and .MBS; eventually, you can modify path and name in the related text box. BROWSE tool helps you to define the path of your project files.

If the defined rule file already exists, a message box will appear asking you whether to overwrite the file or use it as rule file for the current project. If you choose to use it without overwriting, AFM will accomplish a control of correctness to verify the compatibility between the project parameters defined at the previous steps, and the parameters information contained in the rule file.



If the file format is correct, the setup will go on. Otherwise, a message box will appear informing you about the incompatibility of the format file.

The Membership Function file must be a new one. If the selected file already exists, its content will be cancelled.

Using the two drop-down lists on the right side you have to define the number of input/output variables for the current project.

Once stated the choices, click the OK button to go on. Clicking the CANCEL button the FILE NEW procedure will be discarded.

Variables Definition

The Variable Definition dialog box will appear containing:

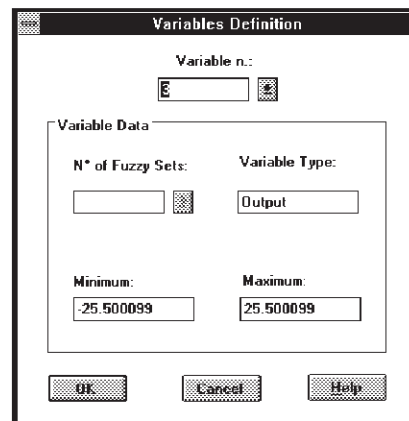
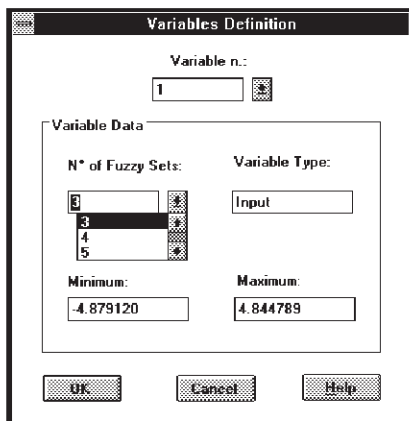
- the drop-down list for the selection of the working variable
- the drop-down list for the fuzzy sets number selection
- a list box displaying the working variable type (Input/Output)
- an edit box for the minimum variable value
- an edit box for the the maximum variable value
- the OK, CANCEL buttons.

For each variable perform the following procedure:

- 1 select the variable using the corresponding drop-down list
- 2 select the related number of fuzzy sets using the appropriate drop-down list.
- 3 fix the minimum and maximum values: the values contained in the edit boxes are directly computed by the program analyzing the pattern file, but you can modify them editing the desired ones.

Once stated the choices, click the OK button to go on. Clicking the CANCEL button the FILE NEW procedure will be discarded. In the case you have selected OK without defining all the variables, a message box will appear informing you of the error occurred. The VARIABLE DEFINITION dialog box will reappear allowing you to complete the procedure.

Note: for the Output variables you have to state only the boundaries. The number of fuzzy sets is directly fixed by the system.



Variables and Fuzzy Sets Renaming

AFM automatically generates I/O variables and related fuzzy set names by using a simple indexing.

ANT_i and $FUZZYSET_{i_j}$ identify the i -th input variable and its j -th fuzzy set respectively; $CONSi$ and $CONSm_n$ for the m -th output variable and n -th fuzzy set for the consequents.

These names could be very useful to understand the behaviour of the fuzzy model. For this reason, a message box will appear allowing you to select the modality to be followed for the fuzzy sets name assignment.

You can choose between:

- automatic assignment: click the NO button. AFM will fix a default name for each fuzzy set.
- manual assignment: click the YES button. A dialog box will appear allowing the user to define a name for each fuzzy set.

Such a dialog box contains:

- the variable number edit box showing the number of the working variable
- the variable name edit box to define the name to be assigned to the working variable
- the fuzzy set number drop-down list to select the fuzzy set of the current variable to which assign the name
- the fuzzy set name edit box to type the name to be assigned to the selected fuzzy set
- a list box containing the names until now assigned to the fuzzy sets of the current variable
- the ADD button which allows to add the stated fuzzy set name in the fuzzy sets names list box
- the DONE button which will be enabled once defined an entire variable.

The screenshot shows a dialog box titled "Insert Names". It is divided into two main sections. The top section has two labels: "Variable:" and "Variable Name:". The "Variable:" label is above a text box containing the number "1". The "Variable Name:" label is above a text box containing the text "Brake_Error". The bottom section has four labels: "Fuzzy Set:", "List Names:", "Fuzzy Set Name (Max 15 char.):", and "Add". The "Fuzzy Set:" label is above a text box containing "2" and a small square icon with a downward arrow. The "List Names:" label is above a list box containing three items: "Negative", "FUZZYSET1_2", and "FUZZYSET1_3". The "Fuzzy Set Name (Max 15 char.):" label is above a text box containing "Zero". The "Add" label is above a button. At the bottom of the dialog are two buttons: "Done" and "Help".

To set the names you have to perform, for each variable, the following procedure:

- 1 the working variable number is automatically displayed in the corresponding edit box
- 2 edit its name in the variable name edit box
- 3 select the working fuzzy set using the corresponding drop-down list
- 4 type the fuzzy set order number in the apposite edit box
- 5 click ADD button to add it in the name list box.
- 6 click DONE button once you have defined all the fuzzy sets names for the working variable.
- 7 Once assigned the fuzzy sets names, a message box will appear informing you that the FILE NEW procedure is finished. The current project is the new one and you are enabled to work on it. In the main window caption the "AFM" name will be replaced with the project name defined at step 1.

Opening an Existing Project

To open an existing project you can use FILE OPEN command.

Everything previously defined will be discarded from memory. If the current project had been modified after the last saving, after selecting OPEN a save request message box will appear. Choosing NO the project will be discarded without being saved, choosing YES it will be saved according to the information defined in the save request dialog box allowing you to go on with the definition of a new project.

The OPEN FILE dialog box will appear, allowing to choose the existing project to be open.

Once selected the project file, the OPEN PROJECT dialog box will appear informing you about the parameters characterizing the problem: target device, number of I/O variables, training and reporting file names, inference method and Mbfs shape.

Click the CONFIRM button to load the project.

The screenshot shows a dialog box titled "Open Project". Inside, there is a section labeled "Project Configuration" with the following fields and values:

- Target: W.A.R.P.-2.0
- Antecedent Number: 2
- Consequent Number: 1
- Pattern File Name: tutor.pat
- Rule File Name: tutor.rul
- Membership File Name: tutor.mbs
- Inference Method: Minimum
- Membership Shape: Scalene Triangles

On the right side of the dialog, there are three buttons: "Confirm", "Cancel", and "Help".

Saving the Project

Choose FILE SAVE in order to save a project into a file. It will be automatically saved with the name appearing in the main window caption.

The SAVE item is dimmed if no project has been loaded or if the current one has been already saved.

In order to define or change the name of the project while saving it into a file, choose SAVE AS. The SAVE AS dialog box will automatically appear (fig.6)

Type the desired name in the FILENAME text-box. Select either a name or a path by using the SAVE dialog box likewise for the OPEN item. Then click the OK button. The project will be saved into the disk, and its name will appear in the main window caption. If a project having the same name already exists, a dialog box will appear allowing you either to overwrite the existing one or not.

If there have been problems during saving operation, or if an invalid path has been specified, an error message will appear.

Note: the project file will contain all the information about the antecedent and consequent variables and the pattern, rules and membership function file paths.

Project Information

After opening or creating a project you can get information on it by choosing FILE PROJECT INFO command. The PROJECT INFORMATION dialog box will appear containing:

- the number of I/O variables and their definition: number of fuzzy sets, minimum and maximum boundaries.
- the list of the files involved in the project: pattern file, rule file, Membership Function file.
- the learning phase parameters: inference method, membership functions shape, number of rules and error reached during the last Build Membership session (for more information on this concepts see Build Membership Function session).

To exit the dialog box click the OK button.

Note: the information displayed in the dialog box are of read only type. You cannot modify any of them.

Project Information			
I/O Numbers		Variable parameters:	
Antecedent Number:	2	Variable Number:	1
Consequent Number:	1	Fuzzy Set Number:	5
		Minimum:	-4.879120
		Maximum:	4.844789
Project Files		Learning Parameters	
Pattern File Name:	tutor.pat	Inference Method:	Minimum
Rule File Name:	tutor.rul	Rule Numbers:	15
Membership File Name:	tutor.mbs	Membership Shape:	Scalene Triangles
		Actual Error:	0.679283
OK		Help	

Printing Project Data Structures

Choose FILE PRINT command to print the fuzzy system related to the current project.

The PRINT dialog box will appear.

The dialog box contains:

- three check boxes to mark the information to be printed (rules, membership functions and simulation results)
- the drop-down list to choose the print quality
- the check box to select the "print to file" option
- the edit box for the selection of the number of copies required
- the OK, CANCEL and SETUP buttons.

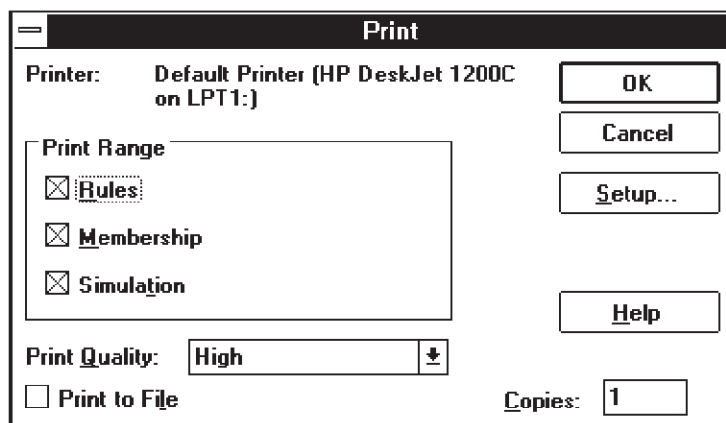
Perform the following procedure to print the information:

- 1 select the data to be printed marking the corresponding check box.
- 2 select the number of copies to be printed.
- 3 click SETUP button to configure the current printer (for more information see Printer Setup command specification.)
- 4 click OK button to go on with the print command.

Note:

The Rules and Membership check boxes are enabled only if they have already been generated using the corresponding BUILD commands.

The Simulations check box is enabled if at least one simulation window is open, and only the opened ones will be printed.



Printer Setup

Choose FILE PRINT SETUP to select and configure the current printer. In order to utilize the setup options refer to the printer manual or to the On Line Help, which is approachable from the same Setup window.

Exit from AFM

Choose FILE EXIT in order to end the current session. The main window will be automatically close and AFM session will finish.

If the current project has not been saved a save request message-box will appear.

Press the YES button in order to save the project before closing the run session, press NO to exit without saving the last changes made.

Learning Commands

AFM allows either to extract or define the inferencing rules related to the project encoded by means of the File NEW command, and next to define and tune the shape and position of the Membership Function shapes associated to each variable.

Note: the Learning commands are dimmed if no project file has been previously loaded.

Build Rules

This menu item has two sub-menus allowing you to choose between the automatic extraction of the rules using neural networks and the manual insertion. Such rules will be used as starting point for the second phase i.e. Build Membership Learning, so a good Mbfs initialization is very important to avoid falling in a local minimum of the error function.

If you already know where to place the centroid of the rules , i.e. the point in which a fuzzy sets has its maximum truth value, you can choose manual insertion and insert them in the related dialog box (see Manual Insertion paragraph for insertion procedure). Instead, use neural network if you want AFM to find such centroids.

Using Neural Net

This sub-menu allows to extract the if-then rules describing the correlation among the variables, in accordance with the fuzzy sets number defined during the creation of the project.

Choose BUILD RULES USING NEURAL NET command in order to carry out the operation. The first time you select the command a dialog box will appear allowing you to choose either to generate all the possible rules, in accordance with the number of fuzzy sets defined over the variables, or to generate only the necessary ones.

Then the BUILD RULES dialog box will appear requiring you the parameters necessary to initialize the Neural Network and perform the learning. It contains:

- an edit box to write the number of iterations to be performed
- an edit box to write the minimum mean error required (called ERROR)
- an edit box to write the minimum mean movement required for the centroids (called DELTA)
- two radio buttons to select the parameters correlated to the learning rate. These buttons are labelled Automatic and Manual. Choosing the Automatic option AFM will set the learning rate value by itself; choosing the Manual option you have to edit in the apposite edit box a learning rate value in the range (0,1). In case of an incorrect value, a message box will appear informing you about the error occurred and a new value will be required. The automatic option is activated by default
- an edit box showing the iteration number, current error and centroids reached during the elaboration
- the RUN, CLOSE and PAUSE buttons
- Error, learning rate, number of iterations and delta parameters play an important role during the learning phase. In fact they affect its termination conditions and the neural network behaviour

Termination conditions

- Error parameter indicates the medium error you want to obtain. When the current error will be smaller than such a value, the learning phase will stop.
- Delta parameter indicates the average of centroid movements. A small current delta value indicates the neural network is becoming stable and to perform further iterations have a slight effect. So, you have to insert a value which will be considered as a threshold. When current delta will reach this threshold, the neural network is considered stable and the learning phase will stop.
- If none of the previous conditions is true, iteration number indicates how many times the neural network has to look over the pattern values .

Neural Network Behaviour

- You can affect the neural network behaviour inserting a learning rate value belonging to the range (0,1). The more such a value is increased the more the neural network learns, but a high learning rate can cause the neural network to never reach the local minimum error. In fact, at each iteration the neural network changes its weights according to the algorithm which uses such a parameter to compute the new weight. So a small learning rate implies a slow convergence to the minimum error, instead a high learning rate can improve such a convergence speed.
- To perform a learning session you have to set all the parameters required and press RUN button. During the operative phase the dialog box is deactivated except for the PAUSE button which allows to interrupt the learning phase before reaching a stop condition.

Once the operative phase has been completed, the dialog box becomes active. You can perform another operative session choosing RUN, or exit the dialog box clicking the CLOSE button. Once the dialog box has been closed, the main AFM windows is restored.

Learning Rate			
<input checked="" type="radio"/>	Automatic	Step:	10
<input type="radio"/>	Manual	Current Error:	6.756687
	<input type="text"/>	Current Delta:	-0.230890

Parameters		
N* of iterations:	Delta:	Error:
<input type="text" value="100"/>	<input type="text" value="0.001"/>	<input type="text" value="0.001"/>

Buttons: Run, Close, Pause, Help

Manual Rules Definition

This sub-menu allows to manually set the position of the center of each fuzzy set instead of letting them to be automatically selected. The inferencing rules, using these values, are generated by AFM itself.

Choose Build Rules MANUAL INSERTION command. The Centroid Insertion dialog box will appear. It contains:

- a drop-down list to select a variable number (working variable)
- an edit box showing the working variable name
- two edit boxes showing the minimum and maximum values of the working variable
- two list boxes showing the working variable fuzzy sets names and the corresponding centroid default value
- an edit box to type the centroid values
- the ADD button to add the centroid values in the apposite list box
- two radio buttons to select between the generation of all possible rules and only activated rules
- the OK and CANCEL buttons.

To set the centroid values you have to perform, for every fuzzy set of each variable, the following procedure:

- 1 select the number of the working variable using the corresponding drop-down list
- 2 select the working fuzzy set using the corresponding list box
- 3 type the centroid value
- 4 click ADD button to add it in the centroid list box.

Once all the fuzzy sets centroid values have been assigned, select the radio button corresponding to the way to be followed for the inferencing rules generation. Then click OK to go on or CANCEL to discard the MANUAL INSERTION procedure.

Note: "All Possible Rule" radio button will be dimmed if you are stating a project for the general target and the maximum possible amount of rules is more than 2^{14}

Build Membership Functions

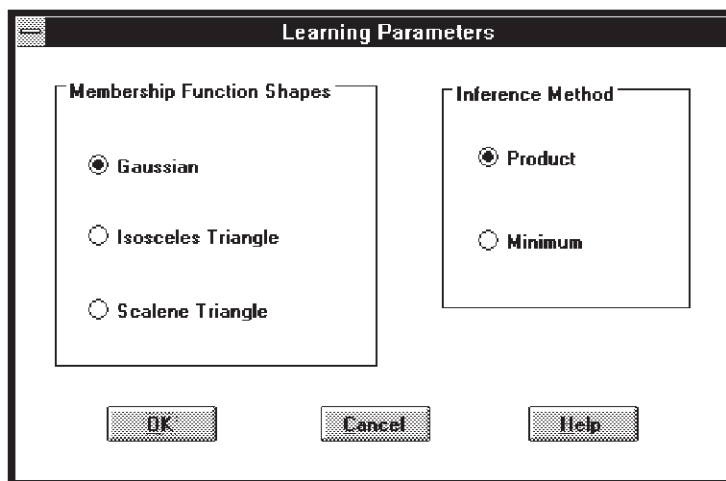
The second step in the fuzzy process modelling consists in the tuning of the shape and position of each membership function associated with the fuzzy sets defined during the creation of the project we are dealing with.

Choose Learning BUILD MEMBERSHIP command in order to carry out the operation.

Mbfs Shape and Inference Method

Before to start the Mbf parameters learning, define which kind of Mbfs - isosceles triangles, scalene triangles or gaussian shape - and which fuzzy intersection operator to be applied during the inferencing phase.

The fuzzy intersection operator represents the way to combine the membership degree of the antecedents of each rule. The most commonly used is the minimum operator; AFM allows to select either the MINIMUM or the PRODUCT operator.



Mbfs Learning

The Mbfs learning phase is managed by the BUILD MEMBERSHIP FUNCTIONS dialog box.

The dialog box contains:

- an edit box to write the number of iterations to be performed
- an edit box to write the minimum mean error required
- three couples of radio buttons to select the parameters related to the learning. The learning rates you have to state are: one for the antecedent fuzzy sets position, one for the antecedent fuzzy sets shape, one for the consequents fuzzy sets position. The buttons are labelled Automatic and Manual. Choosing the Automatic option, AFM will set the learning rates values by itself; choosing the Manual option, you have to edit in the apposite edit box a learning rate value in the range (0,1). In case of an incorrect value, a message box will appear informing you about the error occurred and a new value will be required. Click the OK button to confirm the data and go on with the learning or the CANCEL button to quit the learning phase.
- an edit box showing the iteration number and current error reached during the elaboration
- the RUN, CLOSE and PAUSE buttons.

Note: According to the implemented algorithm, if you have chosen manual learning rate, we suggest you to insert, for the antecedent fuzzy set position and shape, a learning rate whose magnitude order is inferior to that of the consequent fuzzy set positions.

To perform a learning section you have to set all the parameters required and press RUN button. During the operative phase the dialog box is disactivated except for the PAUSE button which allows to interrupt the learning phase before reaching a stop condition.

Once ended the operative phase the dialog box becomes active. You can perform another operative section choosing RUN or exit the dialog box clicking the CLOSE button. Once exited the dialog box the main AFM windows will be restored.

View Commands

The VIEW menu of AFM supplies you with the capability to visualize the fuzzy model extracted for a particular project. It allows a separate visualization of the inference and Membership Function rules.

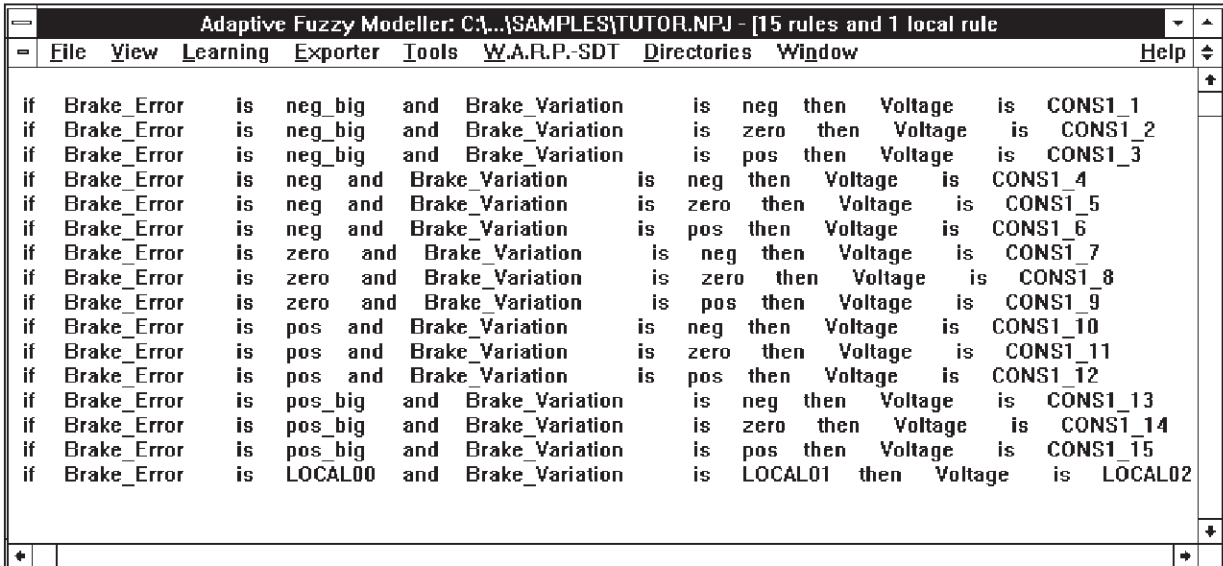
The rules can be visualized in a linguistic format. For the Membership Functions you can choose between a linguistic and a graphical format visualization.

Note: the VIEW commands are dimmed if no project has been previously loaded.

Displaying Rules

In order to visualize the set of rules that AFM has determined for the current project, choose the VIEW RULES command. A dialog box will appear showing the rules list in the following linguistic format:

- each rule is expressed by means of an if-then sentence
- each rule contains antecedents variable fuzzy sets related by an operator *and*. You can use the scroll bars to move inside the rules.



```
if Brake_Error is neg_big and Brake_Variation is neg then Voltage is CONST_1
if Brake_Error is neg_big and Brake_Variation is zero then Voltage is CONST_2
if Brake_Error is neg_big and Brake_Variation is pos then Voltage is CONST_3
if Brake_Error is neg and Brake_Variation is neg then Voltage is CONST_4
if Brake_Error is neg and Brake_Variation is zero then Voltage is CONST_5
if Brake_Error is neg and Brake_Variation is pos then Voltage is CONST_6
if Brake_Error is zero and Brake_Variation is neg then Voltage is CONST_7
if Brake_Error is zero and Brake_Variation is zero then Voltage is CONST_8
if Brake_Error is zero and Brake_Variation is pos then Voltage is CONST_9
if Brake_Error is pos and Brake_Variation is neg then Voltage is CONST_10
if Brake_Error is pos and Brake_Variation is zero then Voltage is CONST_11
if Brake_Error is pos and Brake_Variation is pos then Voltage is CONST_12
if Brake_Error is pos_big and Brake_Variation is neg then Voltage is CONST_13
if Brake_Error is pos_big and Brake_Variation is zero then Voltage is CONST_14
if Brake_Error is pos_big and Brake_Variation is pos then Voltage is CONST_15
if Brake_Error is LOCAL00 and Brake_Variation is LOCAL01 then Voltage is LOCAL02
```

Displaying Membership Functions

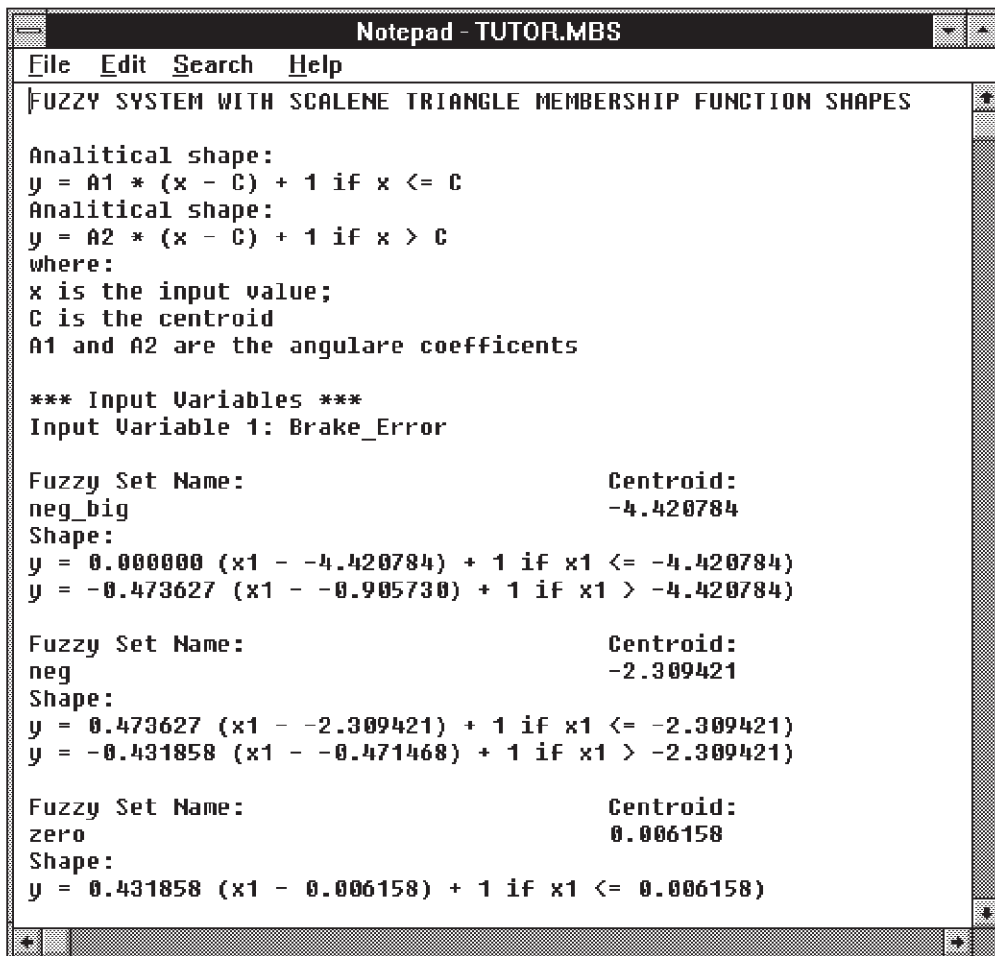
Linguistic Format Visualization

Choose VIEW MEMBERSHIP command. The Notepad will be directly opened displaying the analytical shape for each fuzzy sets of every variable.

The appearing description will differ in accordance with the shape associated to the fuzzy sets of the current project as follows:

- isosceles triangles: such a membership function shape is described using two parameters: the centre, which represents the position of the element having a membership degree equal to 1, and a base size.
- scalene triangle: such a membership function shape is described using three parameters: the centre and the right and left angular coefficient.
- gaussian; such a membership function shape is described using two parameters: the media (i.e. the centre) and the standard deviation.

To exit Notepad-Editor and close it, choose its File Exit command. It is also possible to exit it without closing the Notepad application clicking out of the Notepad window. In this case the main AFM window will be restored.



```

Notepad - TUTOR.MBS
File Edit Search Help
FUZZY SYSTEM WITH SCALENE TRIANGLE MEMBERSHIP FUNCTION SHAPES

Analytical shape:
y = A1 * (x - C) + 1 if x <= C
Analytical shape:
y = A2 * (x - C) + 1 if x > C
where:
x is the input value;
C is the centroid
A1 and A2 are the angular coefficients

*** Input Variables ***
Input Variable 1: Brake_Error

Fuzzy Set Name:                Centroid:
neg_big                        -4.420784
Shape:
y = 0.000000 (x1 - -4.420784) + 1 if x1 <= -4.420784)
y = -0.473627 (x1 - -0.905730) + 1 if x1 > -4.420784)

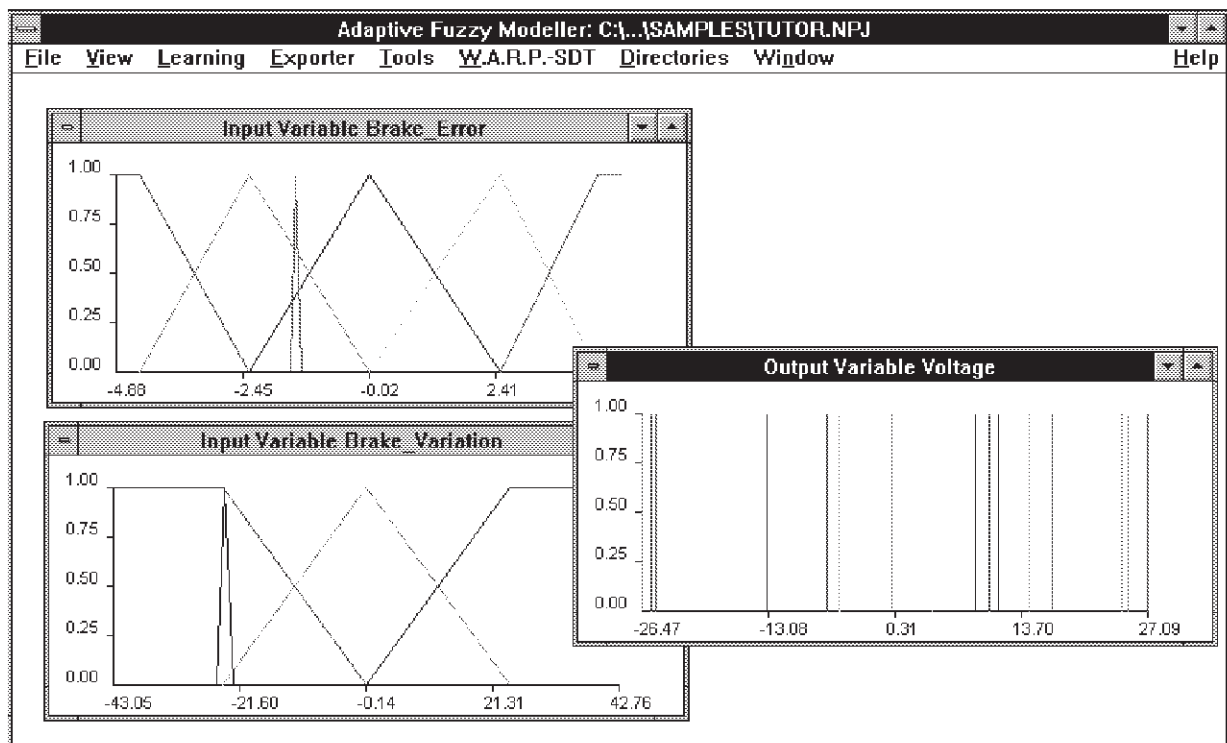
Fuzzy Set Name:                Centroid:
neg                             -2.309421
Shape:
y = 0.473627 (x1 - -2.309421) + 1 if x1 <= -2.309421)
y = -0.431858 (x1 - -0.471468) + 1 if x1 > -2.309421)

Fuzzy Set Name:                Centroid:
zero                            0.006158
Shape:
y = 0.431858 (x1 - 0.006158) + 1 if x1 <= 0.006158)

```

Graphical Visualization

Choose DRAW MEMBERSHIP command. A sub-menu will appear allowing you to select the variable to show. You can choose to display the Membership Functions related to a particular input or output variable, or to display the Membership Functions for all the project variables. For each variable selected you are supplied with a graphical window showing the position and shape of each fuzzy set in the variable domain. You can also display the Membership Function parameter by double clicking on it.



Tools Commands

This menu item is composed by different sub-menus allowing you to manage and test the fuzzy logic knowledge base determined by AFM.

Simulation

The SIMULATION command allows you to carry out the fuzzy computation, i.e. to compute the value inferred for the consequents in correspondence with some input values for the antecedents, using the fuzzy model extracted for the current project.

The simulation can be performed in two ways:

On a single pattern:

allowing you to define the single input pattern values to be used for the simulation.

From File:

allowing you to define the input stream values to use to simulate the system behaviour.

Note: the SIMULATION commands are dimmed if no project has been selected as the current one.

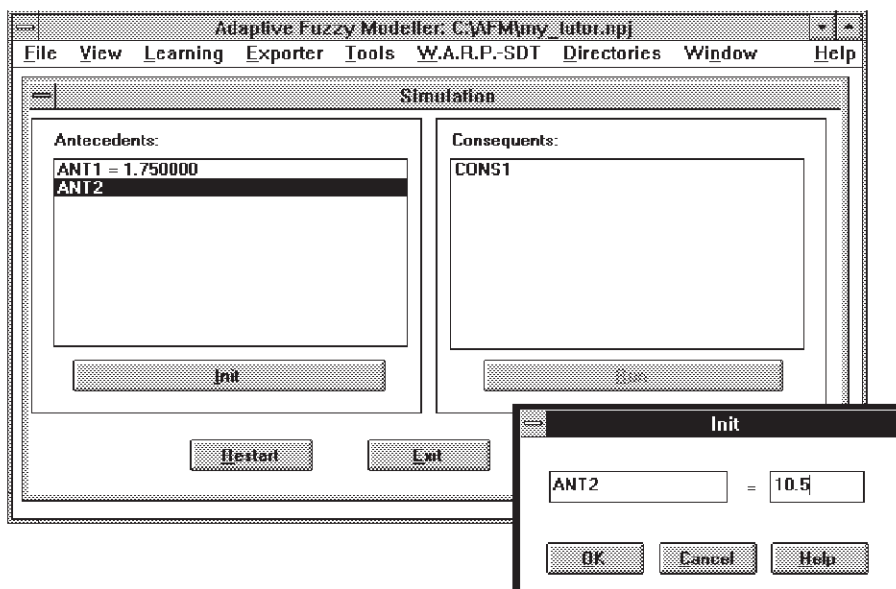
Simulation on a single pattern

Choose TOOL menu and then SIMULATION item. Select SIMULATION command.

The SIMULATION dialog box will appear requiring pattern information. It contains:

- a list box on the left for the antecedent variable names
- a list box on the right for the consequent variable names
- the INIT, RUN, RESTART, EXIT button.

At the beginning all the buttons are dimmed with the exception of the INIT and EXIT ones.



The procedure to carry out in order to perform the simulation from a pattern is the following one:

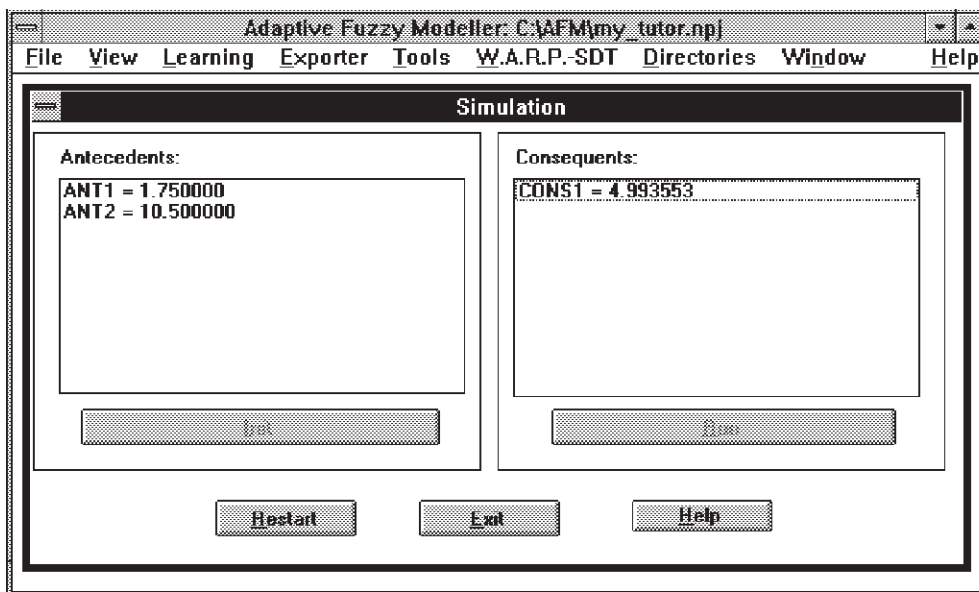
1 antecedent variables initializing:

- select the variable to be initialized clicking on its name in the left list box
- choose INIT in order to go on with the initializing procedure. A dialog box will appear requiring you to supply the variable value. Edit the value in the apposite edit box and click OK button. If the value supplied is out of range, an error message will appear informing you about the incorrect information. In this case you will be required to repeat this step.

Once the value has been fixed, the SIMULATION dialog box will reappear showing the value next to the corresponding variable.

2 Once the entire antecedent variables have been initialized, the RUN button will be enabled. Click it to perform the fuzzy computation and obtain the value inferred on the consequent variables. Such a value will appear next to the consequent names in the right list box of the Simulation dialog box.

To repeat the simulation on a different pattern click the RESTART button and repeat the procedure starting from step 1. To exit the Simulation on a single pattern command, click the EXIT button. The main AFM windows will be restored.



Simulation from File

It allows a simulation for an entire stream of patterns to be performed and supplies you with an output file containing the values inferred on the consequent variables. It can also provide a graphical window showing the comparison between the original function and the result of the simulation obtained using the fuzzy model determined by AFM.

Choose TOOL menu and then SIMULATION item. Choose SIMULATION FROM FILE command to carry out the option. The SIMULATION PARAMETERS dialog box will appear. It contains:

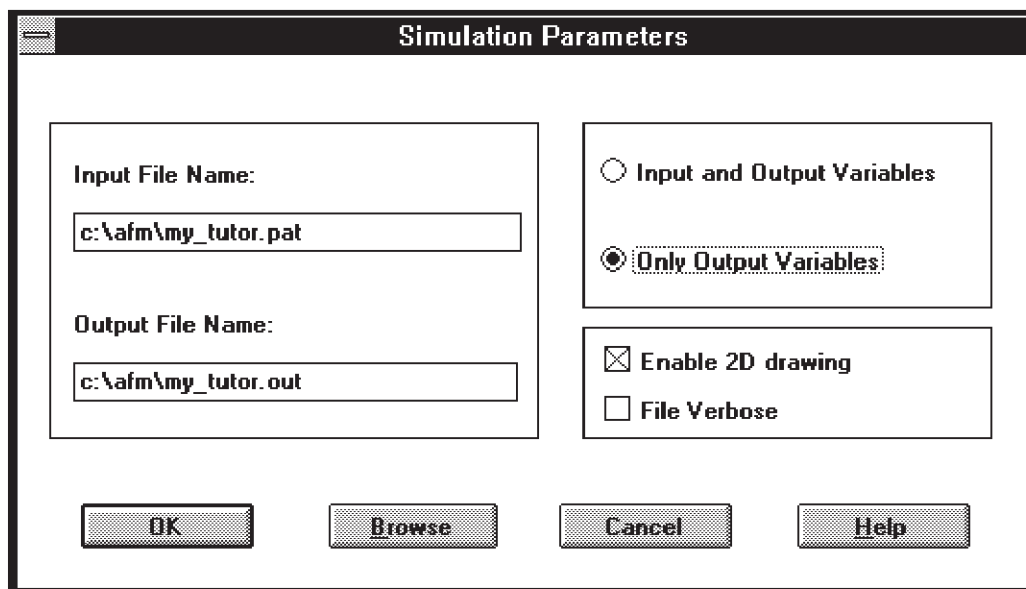
- an edit box for the stream file selection
- an edit box for the output file selection
- two radio buttons to set the output file format
- two check buttons for the 2D drawing enabling and for the file verbose option selection
- the OK, BROWSE and CANCEL buttons.

To discard the Simulation FROM FILE command click CANCEL.

To run the SIMULATION FROM FILE command:

- 1 Select the Input and Output files using the BROWSE button.
- 2 Set the options desired and click OK to go on.

Once the pattern file elaboration has ended, a message box will appear requiring you either to display the output file contents or not. Choosing YES, the Notepad-Editor will be directly opened and the simulation results will be shown. To exit Notepad-Editor and close it, choose its File Exit command. It is also possible to exit it without closing the Notepad application, clicking out of the Notepad window.

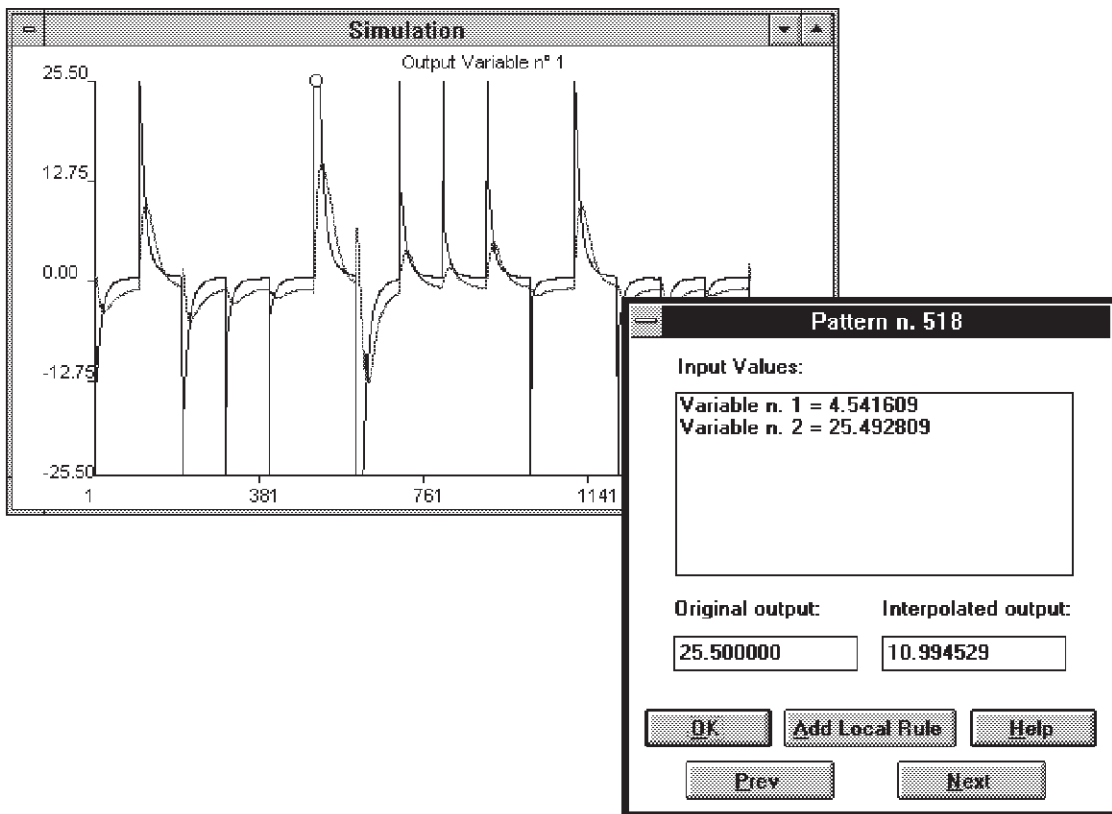


In case you have enabled the 2D Drawing check button, more graphical windows appear (one for each output variable) showing the comparison between the original value (BLUE line) and the simulation result (RED line) plotted pattern by pattern.

When the graphical window is active, you can have information on the single point value by positioning on it using the mouse and double-clicking. A dialog box will appear containing:

- a list box showing the input variables value
- an edit box showing the original output variable value
- an edit box showing the inferred output variable value
- the NEXT and PREV buttons to move among the patterns
- the OK and ADD LOCAL RULE buttons.

Clicking the ADD LOCAL RULE button the input and original output values are used to generate the correspondent local rule which allows to reduce the model error in this specific point. Such a rule can be managed using the Local Rules Command (see Tools Commands). To close the dialog box click OK. Once the local rules have been added, the current simulation results are no more valid and you will be informed by a message box.



Local Rules

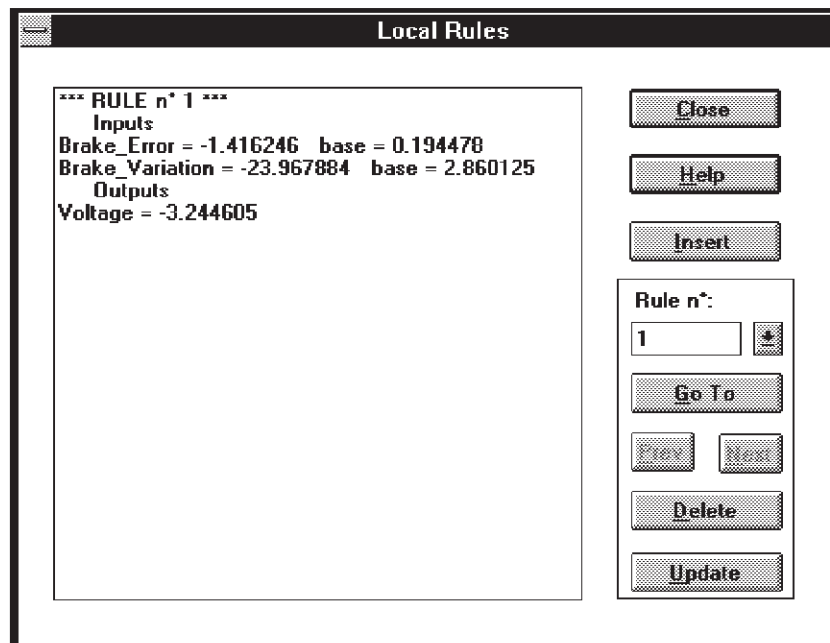
Select TOOLS menu and then choose LOCAL RULES item. The LOCAL RULES dialog box will appear allowing you to view, insert, delete and update the single rules.

The LOCAL RULES dialog box contains:

- an edit box showing the parameters describing the current rules
- a drop-down list to select the number of the rule to be managed
- the PREV, NEXT and GOTO buttons to move among the rules
- the INSERT, UPDATE and DELETE buttons to manage with the rules.
- the CLOSE button to exit the dialog box

Visualizing a Rule

- 1 Select the rule number desired by using the Rule Number drop-down list
- 2 Click the GOTO button. The selected rule will be displayed in the left edit box in the following format:
 - the rule number on the first line
 - the word INPUTS on the second line
 - on the following lines the fuzzy sets values corresponding to the antecedent variables, i.e. the center of the membership function and its base size (these fuzzy sets can be only isosceles triangles).
 - the word OUTPUTS
 - on the following lines the fuzzy sets value corresponding to each consequent variable, i.e. the center of the membership function.



Inserting a New rule

Click the INSERT button. The NEW RULE dialog box will appear allowing you to edit the output values desired for the input values configuration selected.

The NEW RULE dialog box contains:

- a list box on the left for the ANTECEDENTS variables names
- a list box on the right for the CONSEQUENTS variables names
- the INIT ANTECEDENT, INIT CONSEQUENT, OK and CANCEL buttons.

To insert a new rule, initialize all the variables as follows:

- 1 select the variable to be initialized by clicking on its name
- 2 choose INIT ANTECEDENT if the selected variable is an input one or INIT CONSEQUENT if it is an output one.

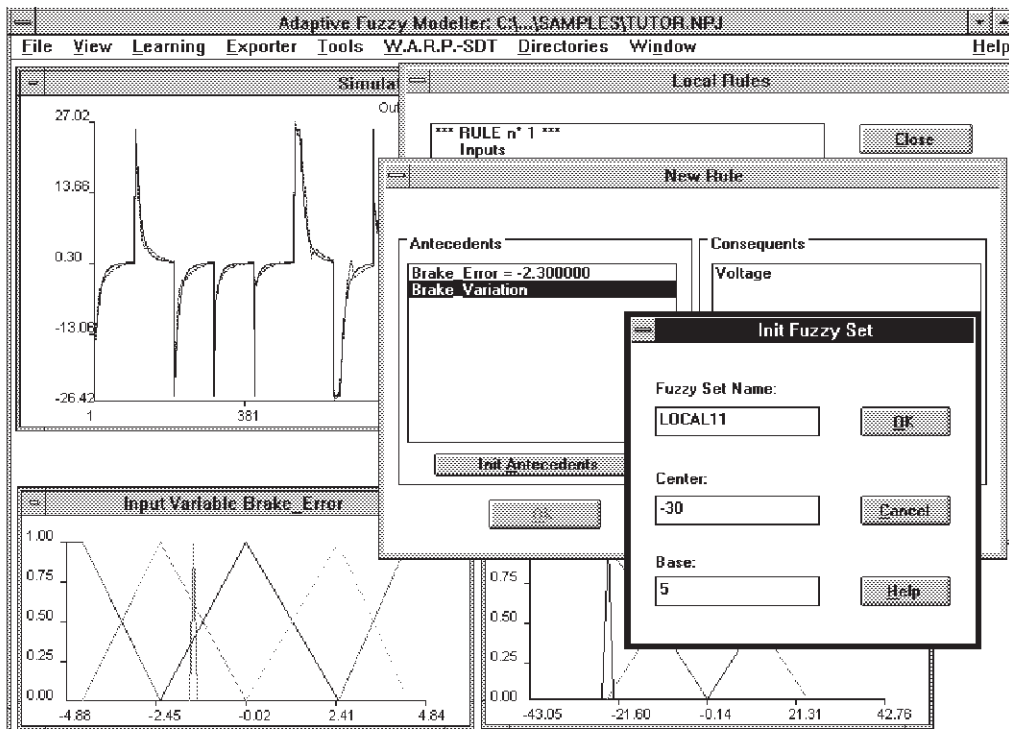
The INIT FUZZY SET dialog box will appear asking you to supply respectively the center, the base size and the name of the antecedent's fuzzy sets; the center and the name of the consequent's fuzzy sets.

Edit the values in the apposite edit box and click OK.

If the supplied value is out of range, an error message will appear warning of the incorrect information. In this case it is required to repeat the step.

Note: if no names are specified, new default fuzzy sets names for this rules are automatically generated.

Once fixed all the variables values choose OK. The Local Rules dialog box will appear again showing the new rule in the left list box.



Deleting a Rule

- 1 Select the number of the rule to delete by using the drop-down list.
- 2 Click the DELETE button. A message box will appear again asking you whether to go on with the rule deleting or not.
- 3 Click YES to delete, NO to exit the delete option

Updating a Rule

- 1 Select the number of the rule to update by using the apposite drop-down list.
 - 2 The initializing values dialog box will appear, allowing you to modify either the antecedent or the consequent values and names in order to modify the rule.
- Once updated the rule values exit the initializing value dialog box by clicking the OK button. The LOCALRULE dialog box will be restored showing the updated rule.

Exporter

The Exporter provides library functions working on the databases automatically generated, which appropriately describe the data structures of the selected project in terms of a different programming environment.

These functions can be exploited inside your programs in order to verify the model extracted and utilize it in real applications.

The sub-menu items provided are:

- To W.A.R.P.1.1
- To W.A.R.P.2.0
- To MATLAB®
- To C Language

Note: the Exporter commands are dimmed if no project has been selected as the current one.

W.A.R.P.-SDT Exporter

Translates the fuzzy knowledge base (rules and membership functions) of a project in a codified project, which can be directly used by FUZZYSTUDIO™ 1.0 W.A.R.P.-Software Development Tool, the W.A.R.P.1.1 development system.

In order to use the W.A.R.P.-SDT Exporter follow these instructions:

- 1 Choose EXPORTER TO W.A.R.P.-SDT command. The OPEN dialog box will appear allowing you to select the destination file which will contain the codified project . In accordance with W.A.R.P.-SDT conventions, the file will be given the extension ".LIZ".
- 2 Click OK button to start the export phase. As soon as it has ended, a message box will appear informing you that the "Building .LIZ file has successful terminated". Click OK to exit the message box and restore the main AFM window.

In case an error occurred during the exporter phase, the command will end without creating the .LIZ file.

Note: for any further information about .LIZ files and W.A.R.P.-SDT, see W.A.R.P.-SDT User manual.

W.A.R.P.2.0 Exporter

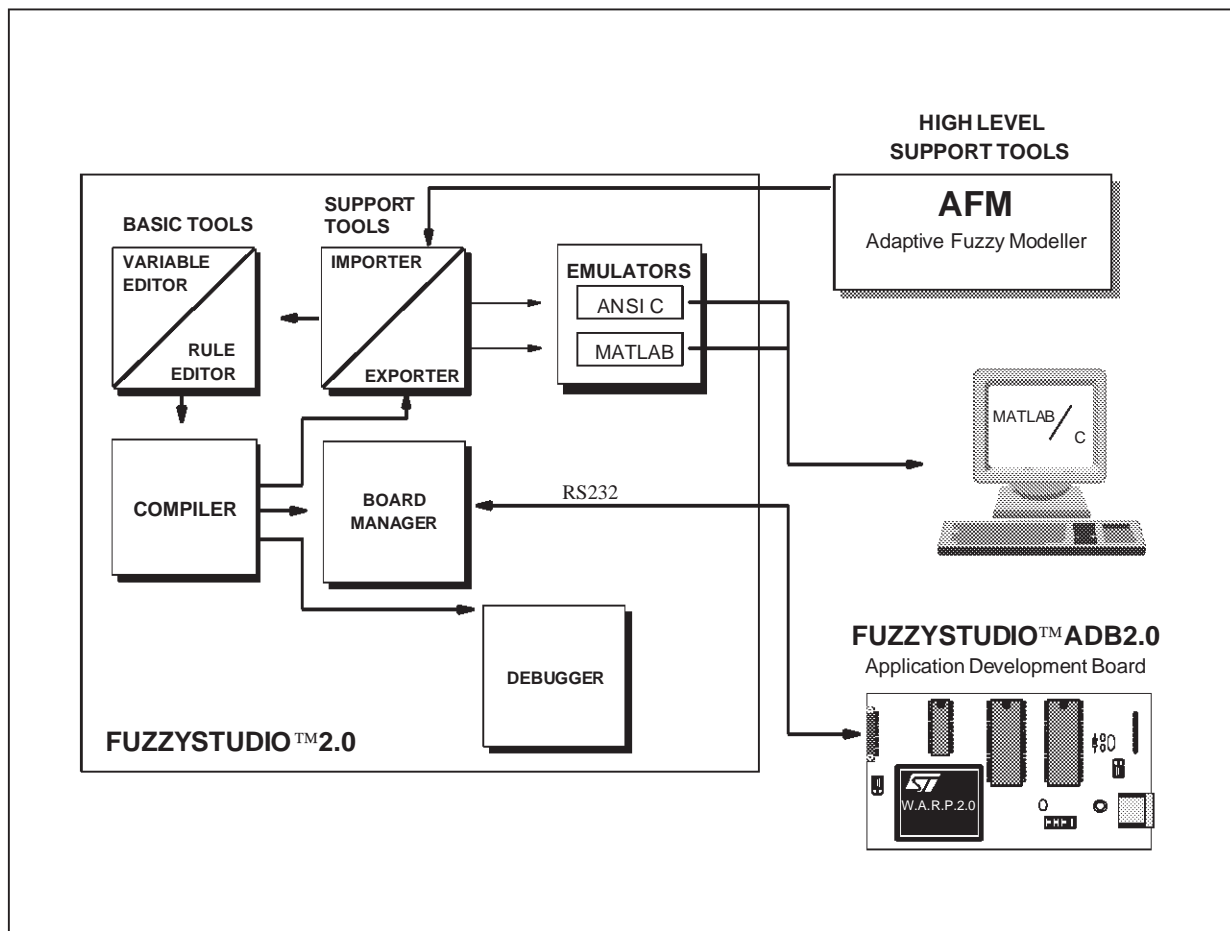
Translates the fuzzy knowledge base (rules and membership functions) of a project in a Fu.L.L. (Fuzzy Logic Language) project format which can be directly used by FUZZYSTUDIO™2.0 i.e. W.A.R.P. 2.0 Development System.

In order to use the W.A.R.P.2.0 Exporter follow these instructions:

- 1 Choose EXPORTER TO W.A.R.P.2 command. The OPEN dialog box will appear allowing you to select the destination file which will contain the codified project. In accordance with FUZZYSTUDIO™ 2.0 conventions, the file will be given the extension ".FUL".
- 2 Click OK button to start the export phase. As soon as it has ended, a message box will appear informing you that the "Building .FUL file has successful terminated". Click OK to exit the message box and restore the main AFM window.

In case an error occurred during the exporter phase, the command will end without creating the .FUL file.

For any further information about .FUL files and FUZZYSTUDIO™2.0, see FUZZYSTUDIO™2.0 User manual.



MATLAB[®] Exporter

Translates the fuzzy knowledge base (rules and membership functions) of a project in a codified project to simulate a control system in MATLAB[®] or SIMULINK environment.

The Exporter provides a .M file containing the data structures necessary to initialize the library function model3.m. This function can be exploited inside the programs developed by the user in order to perform system simulation.

How to use MATLAB[®] Exporter

In order to use the MATLAB[®] Exporter follow these instructions:

- 1 Choose EXPORTER TO MATLAB command. The OPEN dialog box will appear allowing you to select the destination file which will contain the codified project. In accordance with MATLAB[®] conventions, the file will be given the extension ".M" which indicates that the file can be run in MATLAB[®] environment.
- 2 Click OK button to start the export phase. As soon as it has ended a message box will appear informing you that the "Building .M file has successful terminated". Click OK to exit the message box and restore the main AFM window.

In case an error during the exporter phase occurred, the command will end without creating the .M file.

How to use Exporter with MATLAB[®]

To use the so created .M file, enter its name from MATLAB[®] prompt in order to initialize the database. The MATLAB[®] prompt appears and it is possible to list initialized global variables by using MATLAB[®] instructions.

Note: The model.m function must be utilized to simulate the exported project . The input vector is the unique argument of the model3.m function and this is sorted by the same order used for the definition of input variables within the creation of the project. With the MATLAB[®] 4.0 release the model4.m function must be utilized.

How to use Exporter with SIMULINK

SIMULINK graphically defines the control system to be simulated by joining standard function blocks. Each block must be specialized in order to represent a single part of a system to be simulated.

It is possible to associate MATLAB[®] functions created by the user to particular blocks. This operation can be performed by specifying convenient parameters and the model4.m function to the MATLAB[®] function of the Nonlinear library.

In the block specs the function name (here model3.m) and the number of outputs have to be provided. At this point the block can be incorporated inside the project along with the other components in order to simulate the full control system.

To get more detailed information refer to Simulink manual.

C Language Exporter

The C MODEL EXPORTER provides a ".d" file containing an opportune fuzzy data structures coding (if-then rules, Membership Functions values) and an ANSI C source code, named `neural.c`, which implements the inference methods used by AFM.

How to use the C EXPORTER

In order to use the C Exporter follow these instructions:

- 1 Choose Exporter to C MODEL command. A file with ".d" extension is generated, containing the database needed to initialize the internal structure of the library functions.
- 2 During a C program development, it is possible to call three functions included in the libraries. The functions follow these models:

```
int InitNet(char * );  
int PerformModel(float * , float * );  
void FreeNet();
```

The **InitNet** function loads the database included in the file with ".d" extension and initializes the structures used in the *PerformModel* function.

The parameter is a pointer to a string containing the file name with ".d" extension, which eventually includes the complete search-path.

The result of the function is 1 if it has succeeded in loading the structures, otherwise it is 0.

The **PerformModel** function computes the Fuzzy Model output values according to specified input values.

The first parameter is a vector of real number in floating-point precision containing the values of input variables. The second one is a vector of real number in floating-point precision and will contain the values of the output variables computed by *PerformModel*.

The result of the function is 1 if it has succeeded in computing output variables value, otherwise it is 0.

The **FreeNet** function discards all the data structures previously allocated by the *InitNet* function.

```

/*****
/*    STRUCTURE OF A C PROGRAM USING C MODEL EXPORTER    */
/*****
#include.h
#include.h
/*****
/*            FUNCTIONS DECLARATION            */
/*****
extern int  InitNet(char *input_file);
extern int  PerformModel(float *input, float *output);
extern void FreeNet();
/*****
/*            VARIABLES DECLARATION            */
/*****
float Input[NUMBER_OF_INPUT];
float Output[NUMBER_OF_INPUT];
main()
{
.....
        if (!InitNet("SAMPLE.DAT")) exit(0);
.....
.....
/* Load Input Array          */
        if (PerformModel(Input, Output))
/* Normal Execution          */
        else
/* Error Handling            */
/* Print Output Array        */
        FreeNet();
}

```

Directory Management

This menu item allows you to state the default working directories.

Selecting Working Directories

Choosing the DIRECTORY command a dialog box will appear requiring you to state the path for the project_file, the pattern file, the rule file, the Membership Function file.

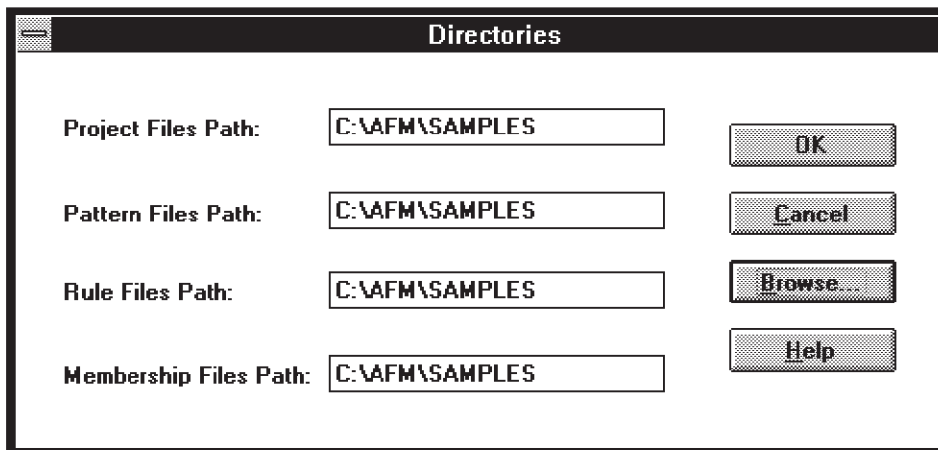
The dialog box will contain:

- 4 edit box, one for each file involved in a project (project file, pattern file, rule file, Membership Function file)
- the OK, CANCEL and BROWSE buttons.

In order to fix a path, select the corresponding edit box and click the BROWSE button. An Open file dialog box will appear enabling you to select the desired path (see "Opening an existing project" paragraph for more information about dealing with the Open file dialog box).

Clicking the OK button the current paths edited in the Directories dialog box will be saved as the current ones, and the DIRECTORY command will end.

The CANCEL button allows to exit the command without saving the last directories changes.



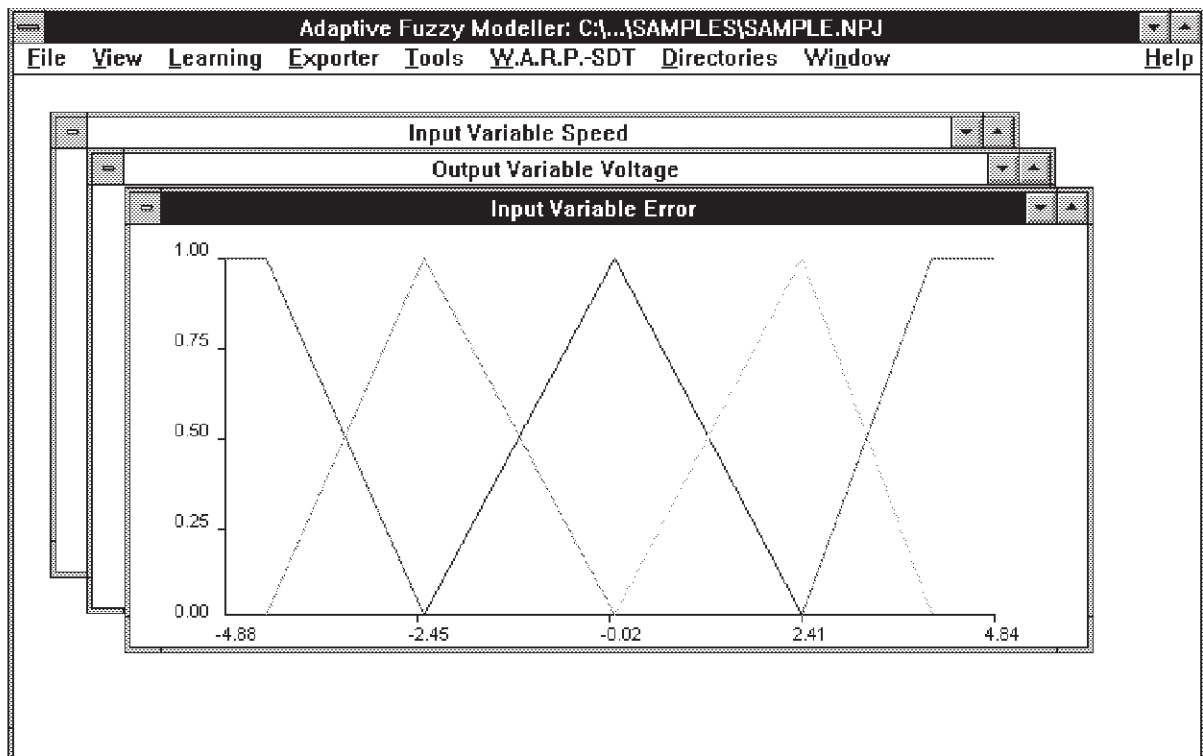
Window Commands

This menu item allows you to manage with the position and size of the graphical windows displayed with the execution of Adaptive Fuzzy Modeller.

The sub-menus supplied are typical of Windows applications. They are:

- Tile: disposes the windows one next to the other
- Cascade: overlaps the open windows keeping the title bar visible
- Minimize All: reduces each windows to an icon
- Close All: closes all the windows

For more information see Windows User Guide.



Files Generated by AFM

In this section you will be supplied with the specs of the files format in order to correctly carry out an AFM running session.

Private Files

AFM Private Files contain all necessary information for the management of a project. By means of such Private Files it is possible to fix the status reached during the Learning step.

They contain important references to the files' absolute paths; therefore it is not possible to move these files from a directory to an other one, or to carry out any kind of change otherwise the project will lose its validity.

The Private Files can be easily recognized from the extension type and they respectively indicate:

.NPJ	Project Parameters
.LPJ	Fuzzy Sets Labels
.MPJ	Mbfs Parameters
.RPJ	Rules Parameters
.LER	Mbfs Learning Rate
.EXT	Local Rules Parameters

Note: The Private Files do not have modified directly by the user, with any kind of word processor. All data contained in the project could be irreparably damaged.

Reporting Files

The Reporting Files allow you to obtain a report of the project, in ASCII format, which can be read by means of a standard Text Editor (i.e. Windows Notepad or DOS Edit). The extension of these files will refer to:

- .MBS** Mathematical description of the Mbfs shape associated to the fuzzy set
- .RUL** List of the project rules

```

Notepad - TUTOR.RUL
File Edit Search Help
Number of Rules: 15
*** Input Variables ***
Input Variable 1: Brake_Error

Fuzzy Set Names:           Centroids:
neg_big                    -4.418377
neg                        -1.549077
zero                       -0.013456
pos                        1.643678
pos_big                    4.375095

Input Variable 2: Brake_Variation

Fuzzy Set Names:           Centroids:
neg                        -24.995188
zero                       0.052569
pos                        24.255243

*** Output Variables ***
Output Variable 3: Voltage

Fuzzy Set Names:           Centroids:
CONS1_1                    -20.719158
CONS1_2                    -24.618107
CONS1_3                     9.015186
CONS1_4                    -5.402810
CONS1_5                    -12.253860
CONS1_6                    15.712183
    
```

```

Notepad - TUTOR.MBS
File Edit Search Help
FUZZY SYSTEM WITH SCALENE TRIANGLE MEMBERSHIP FUNCTION SHAPES
Analytical shape:
y = A1 * (x - C) + 1 if x <= C
Analytical shape:
y = A2 * (x - C) + 1 if x > C
where:
x is the input value;
C is the centroid
A1 and A2 are the angular coefficients

*** Input Variables ***
Input Variable 1: Brake_Error

Fuzzy Set Name:           Centroid:
neg_big                    -4.420784
Shape:
y = 0.000000 (x1 - -4.420784) + 1 if x1 <= -4.420784
y = -0.473627 (x1 - -0.905730) + 1 if x1 > -4.420784

Fuzzy Set Name:           Centroid:
neg                        -2.309421
Shape:
y = 0.473627 (x1 - -2.309421) + 1 if x1 <= -2.309421
y = -0.431858 (x1 - -0.471460) + 1 if x1 > -2.309421

Fuzzy Set Name:           Centroid:
zero                       0.006158
Shape:
y = 0.431858 (x1 - 0.006158) + 1 if x1 <= 0.006158
y = -0.400447 (x1 - 0.004797) + 1 if x1 > 0.006158
    
```

Learning & Testing Files

These files are used either during the Learning and the Testing of the project:

.PAT The format of the PATTERN FILE consists of a line for each pattern, i.e. the Input and Output values of the system to model.

Note: For a verification of the obtained model, during the Testing you are suggested to use the file.PAT with data different from the ones used to train the network. The Output values can be visualized in the SIMULATION window.

.OUT The OUTPUT FILE contains a pattern list relative to the values inferred by the fuzzy model. Then it is possible to use mathematical environments (MATLAB®, MAPLE, etc.) to extrapolate the best performance rate.

This format file differs on the base of the option selected during the simulation procedure. If you have selected I/O radio button, the format file is the same as the previous ones:

- a number of lines equal to the number of patterns
- a pattern for each line
- a number of fields equal to the total number of input and output variables for each pattern stating the input and output values

If you have selected only the output radio button, the format file is:

- a number of lines equal to the number of patterns
- a pattern for each line
- a number of fields equal to the total number of output variables for each pattern stating the output values.

Brake_Error column data

Brake_Variation column data

Voltage column data

Brake_Error column data	Brake_Variation column data	Voltage column data
-1.3994362e+000	0.0000000e+000	-1.3434587e+001
-1.3994362e+000	-2.8776568e-004	-1.3434525e+001
-1.3994360e+000	-1.9955122e-003	-1.3434154e+001
-1.3994316e+000	-2.3516001e-002	-1.3429969e+001
-1.3993081e+000	-2.0027490e-001	-1.3391813e+001
-1.3980483e+000	-9.6554337e-001	-1.3190805e+001
-1.3939414e+000	-2.1809513e+000	-1.2724748e+001
-1.3862757e+000	-3.6391295e+000	-1.2081732e+001
-1.3746985e+000	-5.1053538e+000	-1.1344753e+001
-1.3592829e+000	-6.5371919e+000	-1.0582618e+001
-1.3402824e+000	-7.7970724e+000	-9.8266330e+000
-1.3181199e+000	-8.9364609e+000	-9.1079435e+000
-1.2930619e+000	-9.9890670e+000	-8.4041198e+000
-1.2654690e+000	-1.0837751e+001	-7.7336447e+000
-1.2350087e+000	-1.1565472e+001	-7.1101100e+000
-1.2044340e+000	-1.2126956e+001	-6.5426151e+000
-1.1717524e+000	-1.2560203e+001	-6.0197619e+000
-1.1381245e+000	-1.2838998e+001	-5.5455299e+000
-1.1039101e+000	-1.3011119e+001	-5.1232303e+000
-1.0692843e+000	-1.3160364e+001	-4.7250010e+000
-1.0343868e+000	-1.3205126e+001	-4.3465246e+000
-9.9950523e-001	-1.3151433e+001	-4.0141934e+000
-9.6485455e-001	-1.3035446e+001	-3.7184954e+000
-9.3052281e-001	-1.2917860e+001	-3.4399402e+000
-8.9657754e-001	-1.2734345e+001	-3.1759013e+000
-8.6315862e-001	-1.2526954e+001	-2.9392948e+000

Exporting Files

With AFM it is possible to transfer the fuzzy model to the development systems of SGS-THOMSON Microelectronics W.A.R.P.1.1 and W.A.R.P.2.0. Moreover it is possible to generate a coding in C language or MATLAB® to verify the validity of the model generated by means of a high level simulation.

- .LIZ** Contains the project coding for FUZZYSTUDIO™1.0 W.A.R.P.-SDT development system for the High Level microprocessor fuzzy W.A.R.P. 1.1 (see SGS-THOMSON Microelectronics W.A.R.P. 1.1 data sheet.)
- .FUL** Contains the project coding for FUZZYSTUDIO™2.0 Development System for W.A.R.P.2.0 (see W.A.R.P. 2.0 data sheet for further information).
- .M** It is used as parameter by the procedure model3.m (or model4.m according to MATLAB® version in use, i.e. model4 (PROJECTNAME.M)). This procedure, edited according to MATLAB® language specs, performs the fuzzy computation with reference to the defined project specs, using floating point variables.

Note: If you are using W.A.R.P. 1.1 or W.A.R.P. 2.0 Software Emulator tool, it is necessary to export towards the device and then use FUZZYSTUDIO™ 1.0 or 2.0 (see FUZZYSTUDIO™ 1.0 or 2.0 User Manual).

- .D** The file NEURAL.C contains all necessary procedures for the implementation of an algorithm for fuzzy computation.

The following functions are to be used in C program:

```
int InitNet (char *input.file)
```

Initializes the neural net structure according to the data contained in the file.d, i.e. InitNet("PROJECT.D").

```
int PerformModel (float *input, float *output)
```

The Input values must be memorized within the vector Input, while the Output values are available on the vector Output.

```
void FreeNet (void)
```

It is used to free the memory allocated during the Initialization step. This procedure must be used with care since all neural net parameters could be lost.

Note: If you are using W.A.R.P. 1.1 or W.A.R.P. 2.0 Software Emulator tool, it is necessary to export towards the device and then use FUZZYSTUDIO™ 1.0 or 2.0 (see FUZZYSTUDIO™ 1.0 or 2.0 User Manual).

Appendix B - AFM Tutorial Creation of a Fuzzy Model

This section is a step-by-step guide for the management of AFM tools.

Our target will be to obtain the behaviour of a Fuzzy Controller implemented on a DC Motor driving a servo-mechanism.

A database containing the inferencing rules and the parameters describing the Membership Functions will be generated automatically starting with the system Input/Output sampling. The approach used is based on the Neuro-Fuzzy methodology and will control the position of the motor.

The control has been implemented with a Fuzzy Controller, designed with a traditional algorithm. The final target of the project will be the mapping of the obtained control law in a Fuzzy Controller, which generates the reference value.

Project Definition

The project definition consists of the following steps:

Step 1 Project Name and Target Processor

The project name allows to easily identify all the project files (see "Appendix A" for further information); the Target processor imposes the project constraints according to the different devices.

Step 2 Project Parameters

Learning (.PAT) and Reporting (.RUL,.MBS) file paths definition; I/O Variable number setting.

Step 3 I/O Variables Parameters

Fuzzy Sets number, Universe of Discourse lower and upper bound definition for each variable.

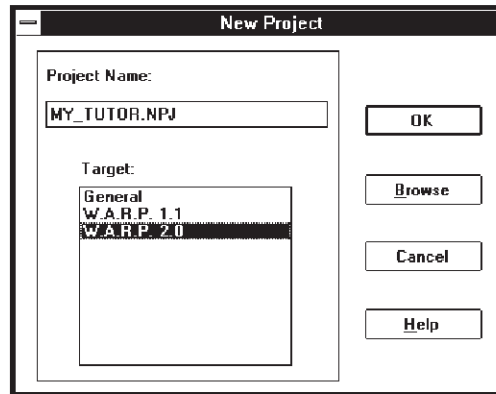
Step 4 Variable and Fuzzy Sets Names

Variable and Fuzzy sets name assignment.

Step 1 Project Name and Target processor

Run AFM double clicking over the icon in the Program Manager, then choose File NEW .

- 1 Insert the project file name **MY_TUTOR.NPJ**.
- 2 Select Target processor **W.A.R.P. 2.0**.
- 3 Click OK button.

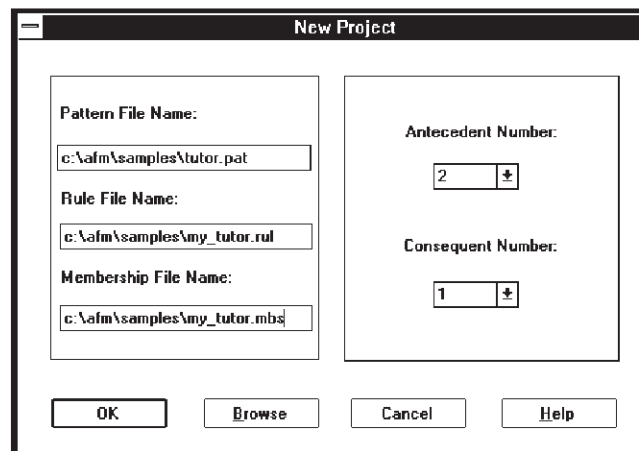


Step 2 Choose Project Parameters

The current project parameters will be defined in the following way:

- 1 Click BROWSE button, then select Pattern file name and its location **TUTOR.PAT**.
- 2 Select Rule file name and its location **MY_TUTOR.RUL**.
- 3 Select Membership file name and its location **MY_TUTOR.MBS** .
- 4 Insert the antecedent and consequent variable number according to the pattern (i.e. controller) structure **ANTECEDENT = 2** , **CONSEQUENT = 1**.
- 5 Click OK button.

Note: Rule File Name and Membership file will be set by default with the same name as your project (my_tutor.RUL) contains the rules of the syntetized Fuzzy Controller. Mbfs file (my_tutor.MBF) contains the variables and the membership functions of the syntetized Pattern File Name, Rule File Name and Membership File Name.



Step 3: Define Variables Parameters

Define range and fuzzy sets number for each variables according to the following:

Brake Error

- 1 Select VARIABLE number 1.
- 2 Choose Variable FUZZY SET number 5.

Brake Variation

- 1 Select VARIABLE number 2.
- 2 Choose Variable fuzzy set number 3.
- 3 Click OK button to confirm the Variables' parameters definition.

Note: a larger number of fuzzy sets gives a more accurate result, but the number of rules increases in linear way.

The screenshot shows a dialog box titled "Variables Definition". At the top, there is a field labeled "Variable n." with the value "1" and a small icon to its right. Below this is a section titled "Variable Data" which contains four fields: "N° of Fuzzy Sets" with the value "5", "Variable Type" with the value "Input", "Minimum" with the value "-4.879120", and "Maximum" with the value "4.844789". At the bottom of the dialog box are three buttons: "OK", "Cancel", and "Help".

Step 4: Variable and Fuzzy Sets Names

A linguistic label is assigned to each variable and fuzzy set to make the project data more comprehensible.

For each variable perform the following procedure:

Input Variable 1

To modify the default Variable name and the related fuzzy set execute:

- 1 Insert **BRAKE_ERROR** in VARIABLE NAME text box .
- 2 Select fuzzy set number.
- 3 Insert the corresponding label **NEG_BIG, NEG, ZERO, POS, POS_BIG** in the FUZZY SET NAME text box.
- 4 Click ADD button after each fuzzy set name insertion.
- 5 Click DONE to confirm the variable parameters and to skip to the next variable.

Input Variable 2

- 1 Insert **BRAKE_VARIATION** in VARIABLE NAME text box .
- 2 select fuzzy set number.
- 3 Insert the corresponding label **NEG, ZERO, POS** in the FUZZY SET NAME text box.
- 4 Click ADD button after each fuzzy set name insertion.
- 5 Click DONE to confirm the variable parameters and to skip to the next variable.

Output Variable 3

- 1 Insert **VOLTAGE** in VARIABLE NAME text box .
- 2 Click DONE to confirm the variable parameters and to leave the Fuzzy Sets default names.

A new project has been created and you can save it using SAVE option in FILE menu.

Learning

The learning phase allows you to automatically extract the inferencing rules (Build Rule), and tune the Membership Functions shape associated to the project (Build Membership).

Build Rules

Choose the item BUILD RULES from menu LEARNING, then select USING NEURAL NET.

To build the rules the following steps have to be performed:

Step 1 Rules Selection Method

- 1 Choose the selection method ALL POSSIBLE RULES radio button.
- 2 Click OK.

Step 2 Set Building Rules Parameter

- 1 Select AUTOMATIC LEARNING RATE radio button.
- 2 Insert NUMBER OF ITERATIONS **100**.
- 3 Insert the DELTA value **0.0001**.
- 4 Insert the ERROR value **0.001**.
- 5 Click RUN button to start the building of the rules.

Note: During the learning phase the dialog box shows you the current iteration number and the current error and centroids movement reached. The learning phase ends when the specified delta value has been reached.

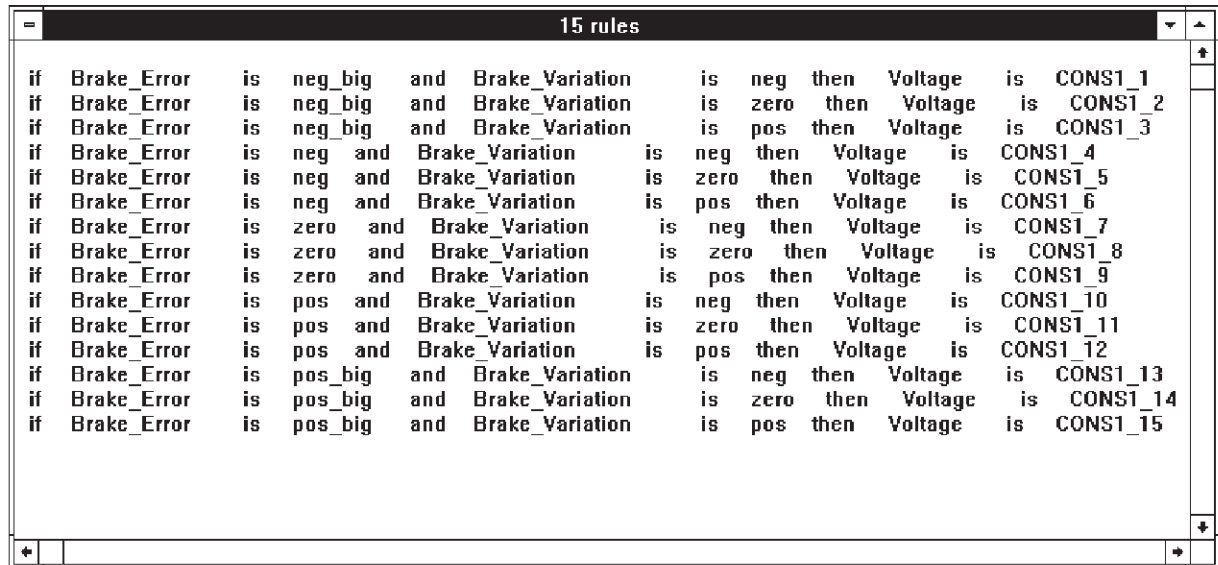
Learning Rate	
<input checked="" type="radio"/> Automatic	
<input type="radio"/> Manual	
<input type="text"/>	

Step:	54
Current Error:	5.269524
Current Delta:	0.024674

Parameters		
N* of iterations:	Delta:	Error:
<input type="text" value="100"/>	<input type="text" value="0.0001"/>	<input type="text" value="0.001"/>

Buttons: Run, Close, Pause, Help

Click CLOSE button and select VIEW RULES to view the built rules.



```
15 rules
if Brake_Error is neg_big and Brake_Variation is neg then Voltage is CONST1_1
if Brake_Error is neg_big and Brake_Variation is zero then Voltage is CONST1_2
if Brake_Error is neg_big and Brake_Variation is pos then Voltage is CONST1_3
if Brake_Error is neg and Brake_Variation is neg then Voltage is CONST1_4
if Brake_Error is neg and Brake_Variation is zero then Voltage is CONST1_5
if Brake_Error is neg and Brake_Variation is pos then Voltage is CONST1_6
if Brake_Error is zero and Brake_Variation is neg then Voltage is CONST1_7
if Brake_Error is zero and Brake_Variation is zero then Voltage is CONST1_8
if Brake_Error is zero and Brake_Variation is pos then Voltage is CONST1_9
if Brake_Error is pos and Brake_Variation is neg then Voltage is CONST1_10
if Brake_Error is pos and Brake_Variation is zero then Voltage is CONST1_11
if Brake_Error is pos and Brake_Variation is pos then Voltage is CONST1_12
if Brake_Error is pos_big and Brake_Variation is neg then Voltage is CONST1_13
if Brake_Error is pos_big and Brake_Variation is zero then Voltage is CONST1_14
if Brake_Error is pos_big and Brake_Variation is pos then Voltage is CONST1_15
```

Build Membership Functions

Choose BUILD MEMBERSHIP item from LEARNING menu and execute the following steps:

Step 1 Set Learning Parameters.

- 1 Choose Mbfs shape [Scalene Triangle].
- 2 Choose Inference method [Minimum].
- 3 Click OK button.

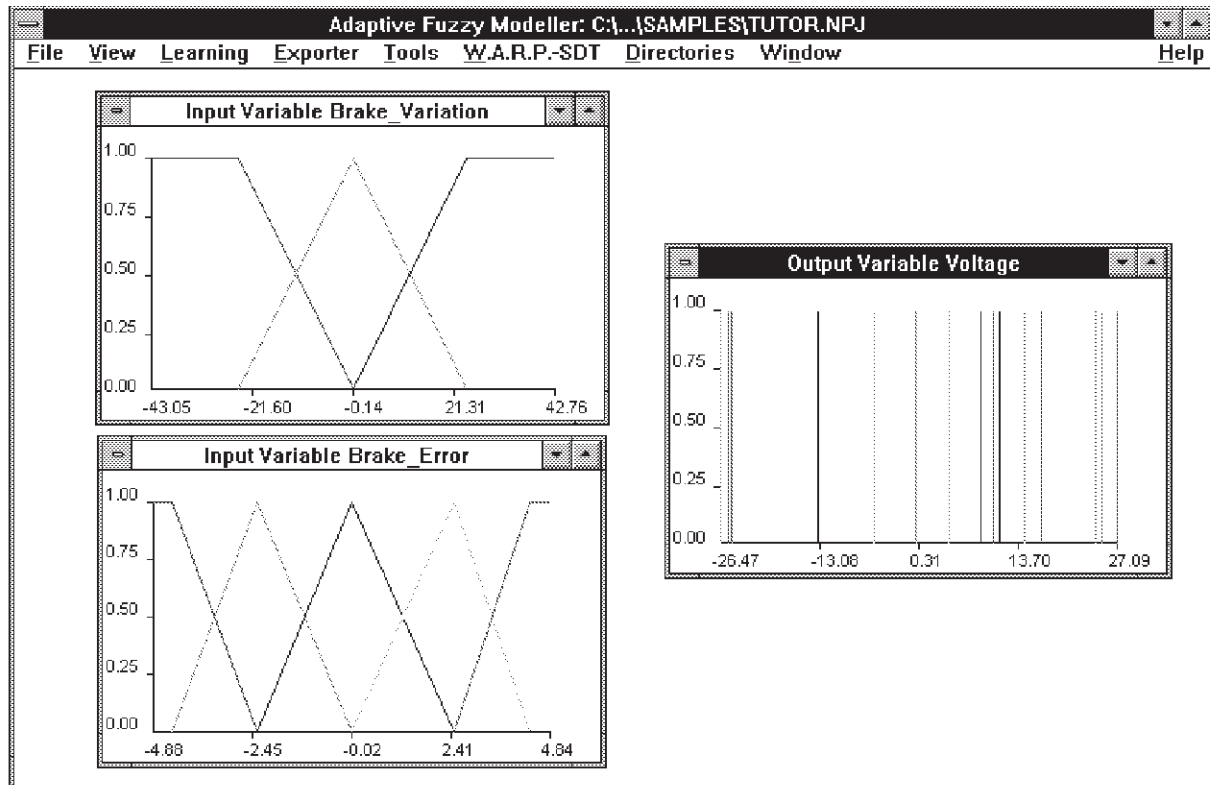
Note: the only available options are the only compatibles with W.A.R.P. 2.0 because this chip has been chosen as target during the definition of the project.

Step 2: Build Membership Functions.

- 1 Choose AUTOMATIC for the learning rates.
- 2 Insert the target error **0.01**.
- 3 Insert the number of iteration to be executed **100**.
- 4 Click RUN button.

If the current error has not achieved the desired result you can run the Learning phase again indicating the number of cycles and clicking the RUN button. If the error is still far from the result, the neural network has probably reached a minimum local point and the automatic learning rates might be too small. In this case it is possible to insert the Learning parameters (the suggested range is between 0.001 and 0.01) in the MANUAL text box to seek the absolute minimum point in a different area.

When the Learning phase is terminated, you can see the obtained Mbf by choosing the DRAW MEMBERSHIP command on VIEW menu (CTRL+A).



Model Simulation

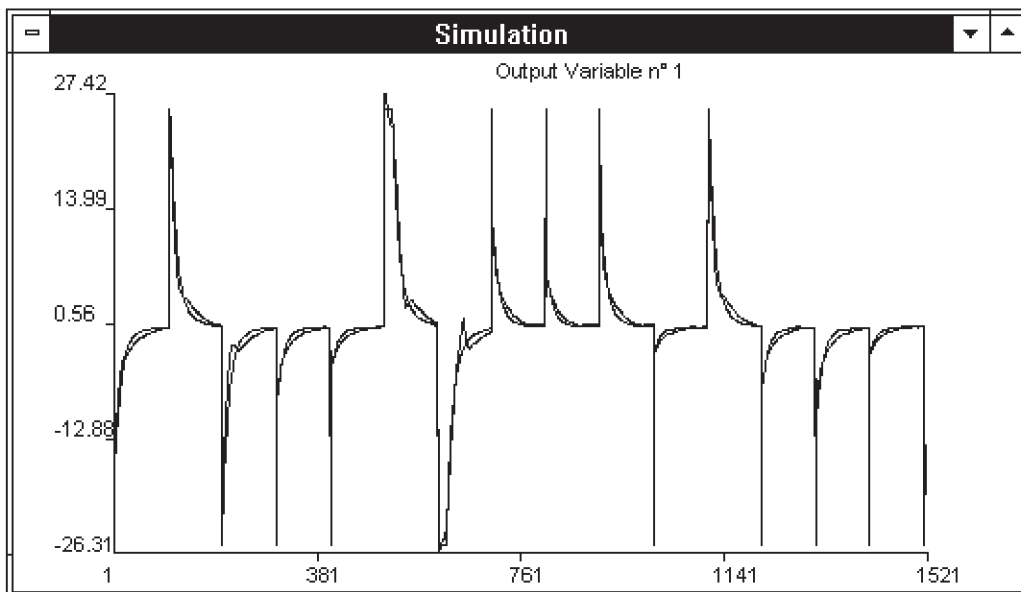
The simulation step allows to carry out the fuzzy computation, i.e. to compute the value inferred for the consequents in correspondence with some input values for the antecedents, using the fuzzy model extracted for the selected process.

From TOOLS menu choose SIMULATION then select SIMULATION FROM FILE:

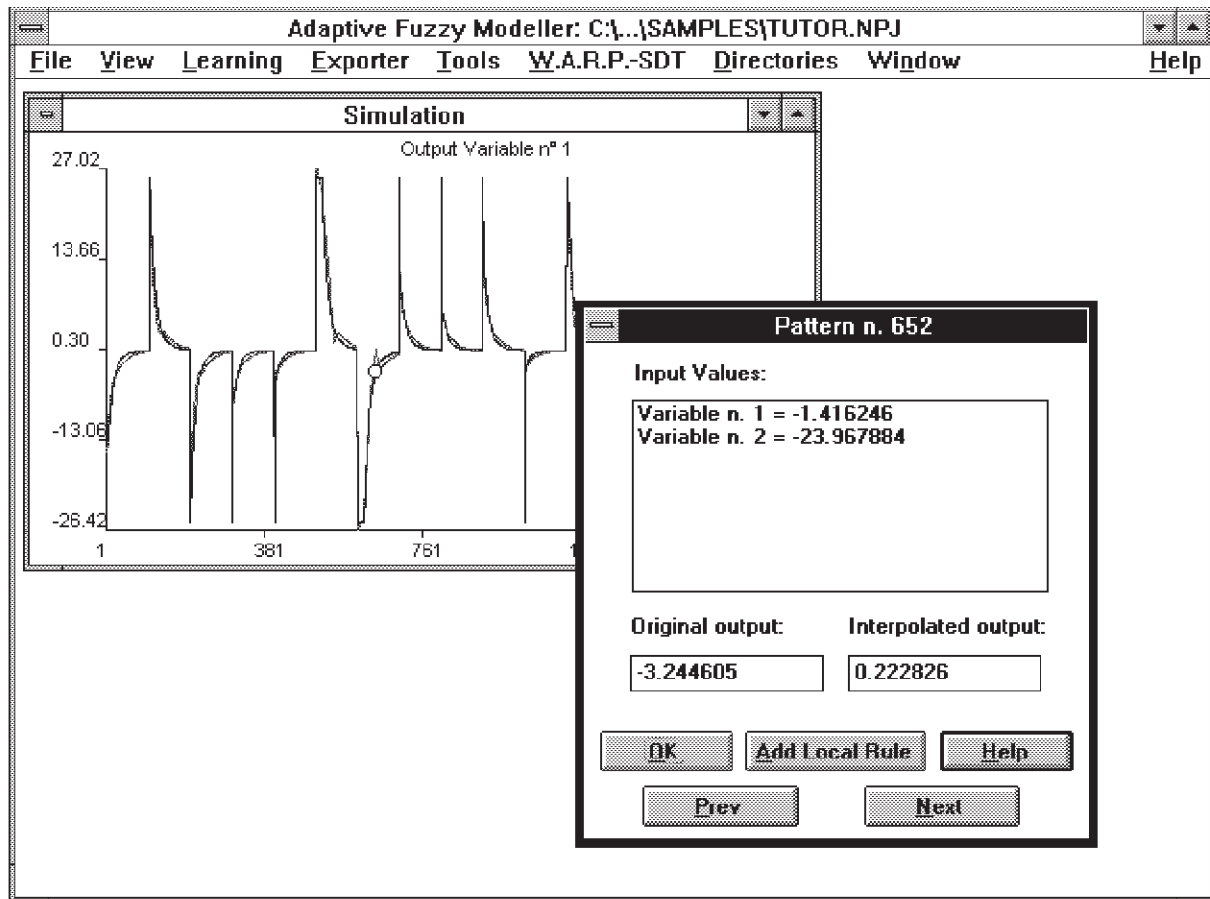
it allows to define the input stream values used to simulate the behaviour of the system. The obtained results can be observed and the zones where the error is still high can be detected.

The following steps have to be executed:

- 1 Choose an INPUT FILE name **TUTOR.PAT**.
- 2 Choose an OUTPUT FILE name **TUTOR.OUT**.
- 3 Check the Enable 2D drawing allowing the plotting of the output.
- 4 Click OK button.



By double-clicking the mouse on the simulation window, it is possible to view the input and the output numerical value of each pattern. It is also possible to add local rules to eliminate error gaps in fixed local points.

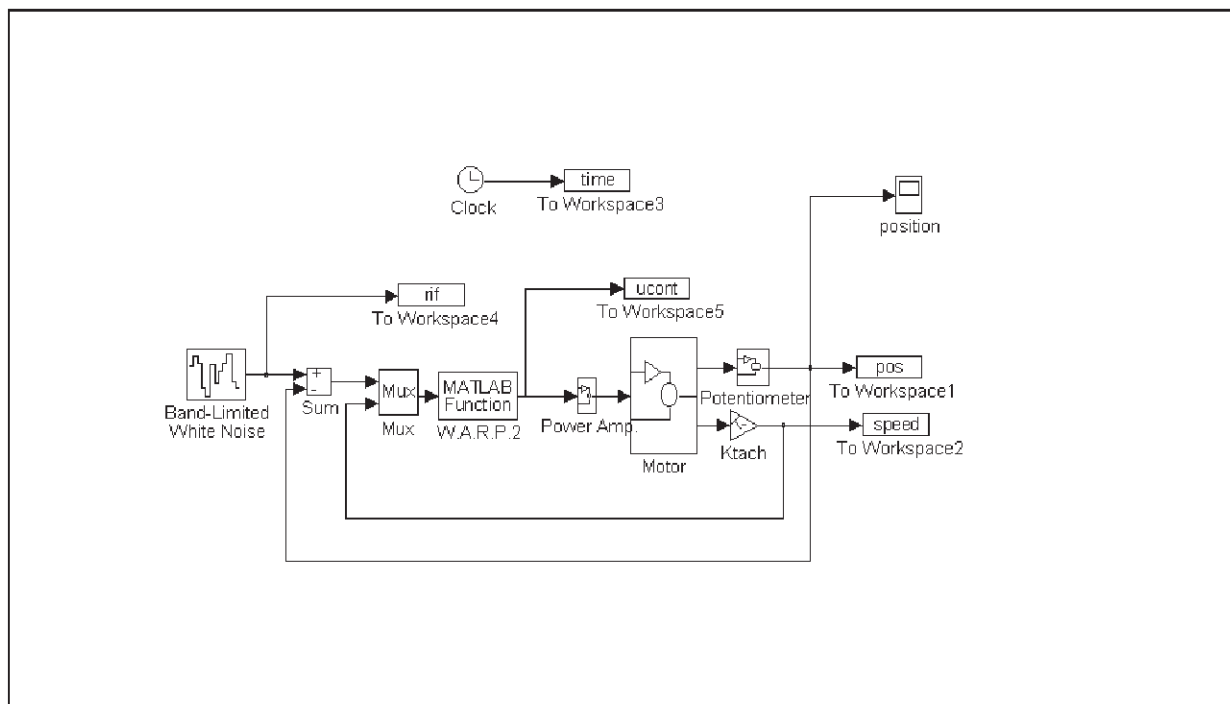


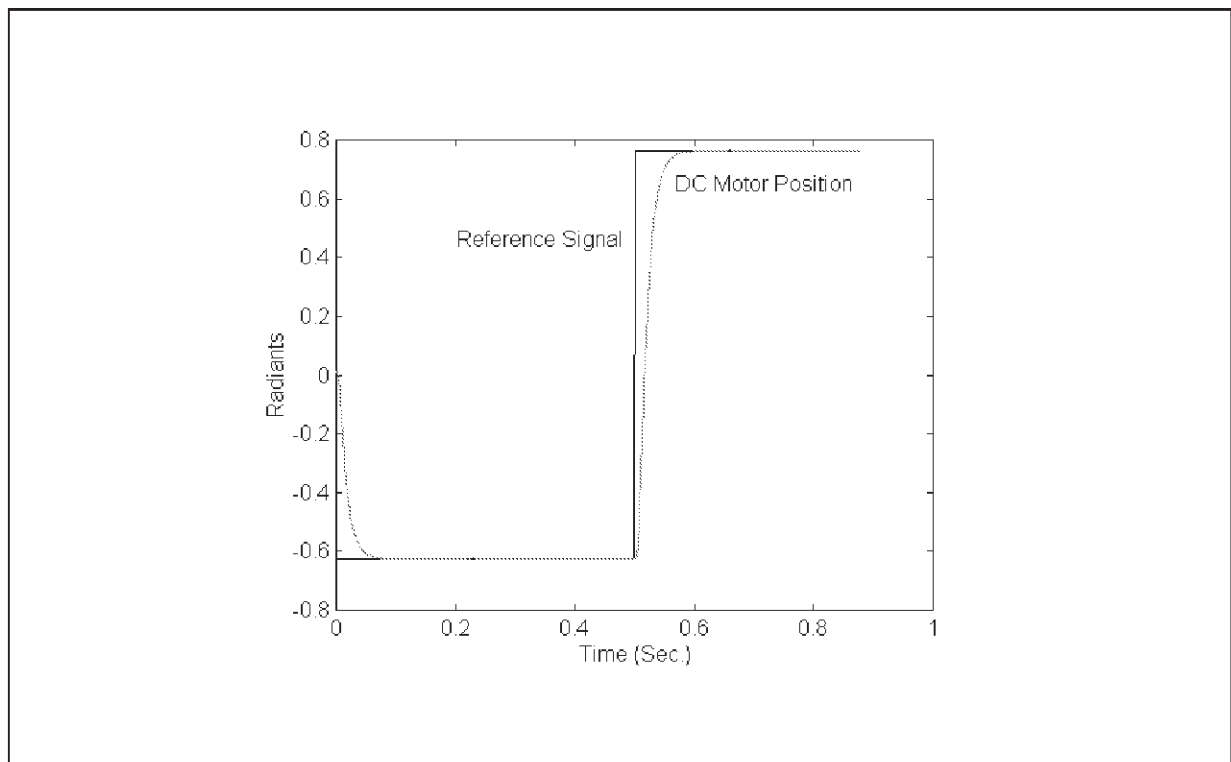
Project Exporting

Exporting to MATLAB®

It is possible to test the efficiency of the controller, performing the simulation of the whole closed-loop system by using a high level simulation environment, like MATLAB®.

- 1 Choose EXPORTER TO MATLAB®.
- 2 Choose output file name and location **TUTOR.M** (.M by default).
- 3 Click OK button.
The result is an .M file (TUTOR.M) which contains all necessary data to represent the fuzzy model. This file will be used by MODEL4.M which is a MATLAB® function implementing the AFM inference. The model4.m function is supplied with the AFM program.
- 4 Run MATLAB® .
- 5 Launch SIMULINK.
- 6 Choose SAMPLES directory by means of the the command: **cd C:\AFM\SAMPLES**.
- 7 From SIMULINK FILE menu select OPEN to TUTOR.M.
- 8 From Matlab prompt, load data for W.A.R.P. 2.0 simulator with the command: SAMPLE.
- 9 Launch the simulation with SIMULINK commands. It is possible to visualize the results by plotting the variables.





Exporting to W.A.R.P.2.0

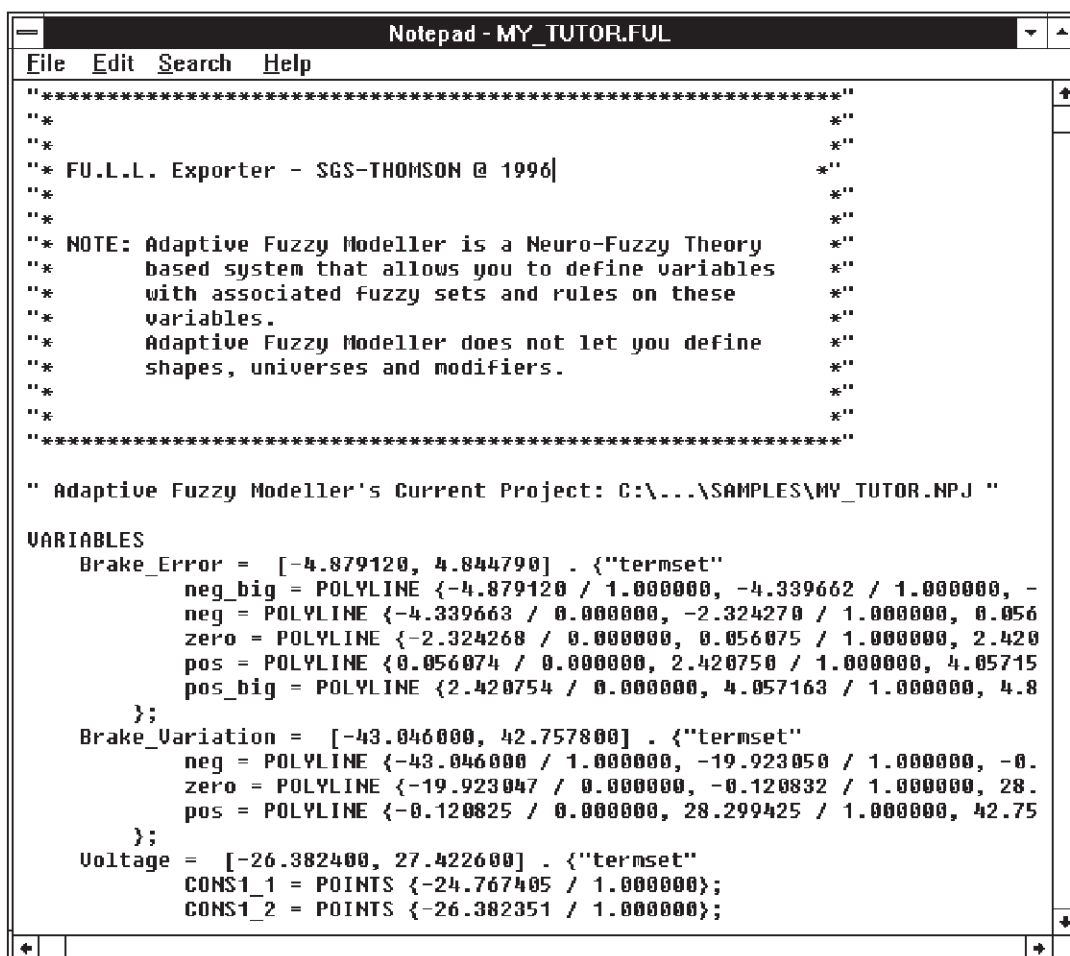
In order to implement the control in the fuzzy device W.A.R.P. 2.0 run EXPORTER, specifying W.A.R.P. 2.0 as target.

The following actions have to be performed:

- 1 Choose TO W.A.R.P.2 from EXPORTER menu.
- 2 Choose the name and the location where to place the output file with .ful as default extension **TUTOR.FUL**
- 3 Click O.K. button.

The result is a .FUL file (TUTOR.FUL) which contains all the data necessary to represent the fuzzy model in the representation language for Fuzzy Logic controller FU.L.L. (Fuzzy Logic Language).

This file will be used in FUZZYSTUDIO™2.0 to import the generated fuzzy model and to translate it in W.A.R.P.2.0 code, then it will be loaded in W.A.R.P.2.0 memories.



```

*****
**
**
** FU.L.L. Exporter - SGS-THOMSON @ 1996
**
**
** NOTE: Adaptive Fuzzy Modeller is a Neuro-Fuzzy Theory
**       based system that allows you to define variables
**       with associated fuzzy sets and rules on these
**       variables.
**       Adaptive Fuzzy Modeller does not let you define
**       shapes, universes and modifiers.
**
**
*****

" Adaptive Fuzzy Modeller's Current Project: C:\...\SAMPLES\MY_TUTOR.NPJ "

VARIABLES
  Brake_Error = [-4.879120, 4.844790] . {"termset"
    neg_big = POLYLINE {-4.879120 / 1.000000, -4.339662 / 1.000000, -
    neg = POLYLINE {-4.339663 / 0.000000, -2.324270 / 1.000000, 0.056
    zero = POLYLINE {-2.324268 / 0.000000, 0.056075 / 1.000000, 2.420
    pos = POLYLINE {0.056074 / 0.000000, 2.420750 / 1.000000, 4.05715
    pos_big = POLYLINE {2.420754 / 0.000000, 4.057163 / 1.000000, 4.8
  };
  Brake_Variation = [-43.046000, 42.757800] . {"termset"
    neg = POLYLINE {-43.046000 / 1.000000, -19.923050 / 1.000000, -0.
    zero = POLYLINE {-19.923047 / 0.000000, -0.120832 / 1.000000, 28.
    pos = POLYLINE {-0.120825 / 0.000000, 28.299425 / 1.000000, 42.75
  };
  Voltage = [-26.382400, 27.422600] . {"termset"
    CONS1_1 = POINTS {-24.767405 / 1.000000};
    CONS1_2 = POINTS {-26.382351 / 1.000000};

```


Full Product Information at <http://www.st.com/stonline/products/support/index.htm>

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specification mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics

© 1996 STMicroelectronics – Printed in Italy – All Rights Reserved

FUZZYSTUDIO® is a registered trademark of STMicroelectronics

DuaLogic™ is a trademark of STMicroelectronics

MS-DOS®, Microsoft® and Microsoft Windows® are registered trademarks of Microsoft Corporation.

MATLAB® is a registered trademark of Mathworks Inc.

STMicroelectronics GROUP OF COMPANIES

Australia - Brazil - Canada - China - France - Germany - Italy - Japan - Korea - Malaysia - Malta - Morocco - The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.
