A MULTIPLE-INPUT-SINGLE-OUTPUT (MISO) FUZZY LOGIC MODEL FOR GENERATION OF WATERSHED SOIL CONSERVATION SERVICE (SCS) CURVE NUMBERS

by

Ebrahim Hosseini

A Thesis Submitted to the
Faculty of Engineering
In Partial Fulfilment of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

Major Subject: Biological Engineering

APPROVED: [redacted]

Dr. Jack Burney

Dr. M. EL-Hawary

Dr. L. Edwards

Dr. A. Madani

Dr. M. Satish

External Examiner

DALHOUSIE UNIVERSITY-DALTECH

Halifax, Nova Scotia 1998

0-612-31526-6

DALTECH LIBRARY

"AUTHORITY TO DISTRIBUTE MANUSCRIPT THESIS"

TITLE:

A MULTIPLE-INPUT-SINGLE-OUTPUT (MISO) FUZZY LOGIC MODEL FOR GENERATION OF WATERSHED SOIL CONSERVATION SERVICE (SCS) CURVE NUMBERS

The above library may make available or authorize another library to make available individual photo/microfilm copies of this thesis without restrictions.

Full Name of Author: Ebrahim Hosseini

Signature of Author [redacted]

Date: May/15/ 1998

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

Abbreviation	Description
ALSO	Logical operator (∨ or ∪)for union in multi-input
AND	Logical operator for intersection (∩ or ∧)
ANN	Artificial Neural Network
AMC	Antecedent Moisture Content
DISO	Double Input Single Output
C10	First Membership for Cover density
C20	Second Membership for Cover density
C30	Third Membership for Cover density
C40	Fourth Membership for Cover density
C50	Fifth Membership for Cover density
C60	Sixth Membership for Cover density
C70	Seventh Membership for Cover density
C80	Eighth Membership for Cover density
C90	Ninth Membership for Cover density
C100	Tenth Membership for Cover density
CN	Curve Number
CN10	First Membership for Curve Number
CN20	Second Membership for Curve Number
CN30	Third Membership for Curve Number
CN40	Fourth Membership for Curve Number
CN50	Fifth Membership for Curve Number
CN60	Sixth Membership for Curve Number
CN70	Seventh Membership for Curve Number
CN80	Eighth Membership for Curve Number
CN90	Ninth Membership for Curve Number
CN100	Tenth Membership for Curve Number
COG	Center of Gravity
DRY	Dry soil
ES	Expert Systems
FAM	Fuzzy Associated Matrix
FC	Field Capacity
FLC	Fuzzy Logic Control
F(t)	Infiltration capacity at time t
HD	Height Defuzzification for all membership function
I_a	Initial abstraction
IMP10	First Membership for Impermeability

IMP20	Second Membership for Impermeability
IMP30	Third Membership for Impermeability
IMP40	Fourth Membership for Impermeability
IMP50	Fifth Membership for Impermeability
IMP60	Sixth Membership for Impermeability
IMP70	Seventh Membership for Impermeability
IMP80	Eighth Membership for Impermeability
IMP90	Ninth Membership for Impermeability
IMP100	Tenth Membership for Impermeability
LC	Low Cover density
LCN	Low Curve Number
LIA	Low Impermeable Area
LS	Low Sandy Soils
M10	First Membership for Moisture content
M20	Second Membership for Moisture content
M30	Third Membership for Moisture content
M40	Fourth Membership for Moisture content
M50	Fifth Membership for Moisture content
M60	Sixth Membership for Moisture content
M70	Seventh Membership for Moisture content
M80	Eight Membership for Moisture content
M90	Ninth Membership for Moisture content
M100	Tenth Membership for Moisture content
MC	Medium Cover density
MCN	Medium Curve Number
MIA	Medium Impermeable Area
MISO	Multi-Input Single Output
Moist	Medium moisture contents
MOM	Mean-of-Maxima
MS	Medium Sandy soils
OR	Logical operator for union ($\cup$)
Q	Runoff potential (mm)
P	Rainfall (mm)
P(t)	Rainfall intensity at time t
Ri	Rule number
S	Maximum potential difference between rainfall and runoff
S10	First Membership for Sand percentage
S20	Second Membership for Sand percentage
S30	Third Membership for Sand percentage
S40	Fourth Membership for Sand percentage
S50	Fifth Membership for Sand percentage

S60	Sixth Membership for Sand percentage
S70	Seventh Membership for Sand percentage
S80	Eighth Membership for Sand percentage
S90	Ninth Membership for Sand percentage
S100	Tenth Membership for Sand percentage
SATU	Saturated Soils
SCS	Soil Conservation Service (U.S.A)
SISO	Single Input Single Output
UoD	Universe of Discourse
VDRY	Very Dry Soil
VHC	Very High Cover density
VHCN	Very High Curve Number
VHIA	Very High Impermeable Areas
VHS	Very High Sandy Soils (% Sand >> % Clay)
VLC	Very Low Cover density
VLCN	Very Low Curve Number
VLIA	Very Low Impermeable Area
VLS	Very Low Sandy soils
W10	First Membership for Woodland
W20	Second Membership for Woodland
W30	Third Membership for Woodland
W40	Fourth Membership for Woodland
W50	Fifth Membership for Woodland
W60	Sixth Membership for Woodland
W70	Seventh Membership for Woodland
W80	Eight Membership for Woodland
W90	Ninth Membership for Woodland
W100	Tenth Membership for Woodland
X^M	Input number for i=1,...4.
${X^M}_N$	Membership function for N = 1,...5.
Z	Output space with membership function Oj for j =1,...5.
$\mu \check{R}_1 \circ \check{R}_2$	Membership function for the fuzzy relation
$\mu_{\hat{A}}(x)$	Membership function for set A[ ] for $\mu_{\hat{A}}(x)$ =[(0,...,1)]
$\mu_B(x)$	Membership function for set B[ ] for $\mu_B(x)$ =[(0,...,1)]
$\mu_{\epsilon\hat{A}}(x)$	Complement or negation of set A

ACKNOWLEDGMENTS

The author wishes to thank to his supervisor Dr. Jack R. Burney for his supervision of this work. My thanks also to other members of my academic guidance committee, and, in particular, to Dr. Mohamad El-Hawary for his many hours of counsel at various stages of the work.

I hereby thank the Minister of Jahad-E-Sazandagi (Minister of Construction) for providing full financial support, and the Minister of Higher Education of Iran for his encouragement of this study program.

Finally, thanks to my wife Maryam and my daughters Najmeh and Nassim, and my son Navvab for their patience and understanding, and their support during the long period of this project.

ABSTRACT

The procedure presented builds a bridge between computational intelligence and hydrologic modeling in applying fuzzy and multivariable data to design in urban and rural watersheds. Watershed parameters such as soil moisture content, cover density and impermeability are viewed as continuous parameters using fuzzy set theory and are then analyzed in a fuzzy logic control expert system. This logic model, involving relations between these parameters uses methods that explicitly take vagueness into account. The theory of fuzzy sets, especially fuzzy modeling, is employed in a new way to represent watershed parameter relations as a set of fuzzy rules. This approach was implemented as an interactive modeling system, called the "Fuzzy Logic Expert Watershed Curve Number" ($FLEW^{CN}$). This model is based on logical relationships between parameters rather than experimental data. Fuzzy logic control methodology was used in development in three consecutive phases in which each phase produced a base for the next phase of development. These phases were firstly, a single-input-single-output (SISO) model for soil texture (% sand versus % clay), followed by double-input-single-output (DISO) models for % sand versus each individual watershed parameter. The final stage of fuzzy logic control was to develop a multi-input-single-output (MISO) model based on the parallel algorithm by forward reasoning strategy with regard to conditional and unconditional rule-based systems. The final model has an open-loop control structure (output has no effect on input).

The program is organized as a procedural process with elements of action (membership functions). Each membership function is defined in binary format as an object of an element. Each element has two different references as a pivot (only a specific element appears to act when the program is called) and as a global (all of the elements are affected when the program calls the elements). This process appears during execution time as a user interface.

The user interface links keyboard input of fuzzy data to the fuzzy control and graphically illustrates each input space. The user interface also indicates the inference action in a screen output space. The user interface provides an effective means to assess the effect of changes

in combination of inputs on the output response.

The program is coded in the Turbo C environment on a DOS platform using the Borland graphic support system. This program is independent of any expert shell during the time of execution.

Verification was carried out by comparison with the Soil Conservation Service (SCS) method. The validation showed that the fuzzy logic model predicted a curve number in a similar range to that of the SCS method. However, predictably some differences were observed which can be attributed to fuzzy logic methodology. This is because fuzzy logic control produces a continuous model, whereas the SCS model is a discrete model. The advantage of the FLEW^{CN} model is that it is not limited to a specific range of inputs; any combination of inputs (%sand, % cover, % moisture, and % of impermeable area) can be translated to an output response (CN).

CONTRIBUTION TO KNOWLEDGE

The main contribution of this work is in building a bridge between computational intelligence and hydrological modeling by using fuzzy set theory to develop a model called Fuzzy Logic Expert Watershed Curve Number ($FLEW^{CN}$). The model converts watershed parameters' continuity into a continuity of curve numbers.

The model is efficiently written on a DOS platform to be used by watershed developers to estimate direct runoff based on the SCS curve number methodology.

The program is based on fuzzy logic control methodology and uses logical interpretation of complex multivariable parameters in a watershed to quantify fuzzy data.

The model is an integrated parallel algorithm and uses a parallel rule firing system with a forward chaining strategy in mixed with parallel rule firing method by applying antecedent rules in conditional and unconditional forms. This strategy gives the program an ability to apply unequal weight rules. An interface, developed for the multiple inputs, visually presents the fuzzy logic membership functions on the monitor screen by use of three connected algorithms that represent initialization, connection and visualization.

1. INTRODUCTION

1.1 Background to Hydrologic Modelling

A watershed is one of the most complex subjects to study as it comprises an interaction between climate, cover complex (vegetation), soil and human activities. Several approaches have been suggested to model these phenomena by physical or mathematical (empirical and theoretical) methods. Nash (1988) pointed out that hydrology has been unable to provide general scientific procedures for predicting the effect of ground cover on hydrological response.

Given the complexity of the system, no model has become widely accepted, for many of the following reasons:

(i) Models are generally limited by the knowledge of the developer,

(ii) Some models are structured by simplifying the concepts and assumptions which then become the basis of model validation,

(iii) Models are based on inadequate casual theory of the physics of hydrologic processes in a watershed,

(iv) Natural phenomena are often defined by a mathematical model without consideration of the relevant features,

(v) Spatial and temporal variability of parameters are not usually accounted for, and

(vi) Some models are not able to be used for management in a large watershed.

For many of the above reasons several models have recently been developed or modified to suit a specific region or application.

Traditional models may be classified in one of the following application categories :

(i) Research models. These models attempt to define system behaviour and formulate principles. These types of studies are governed by the desire to seek fundamental knowledge of modelling as a mathematical representation of a watershed.

(ii) Operational and management models. These models concentrate on the effects of development or operating changes on watershed response.

Recent developments have taken advantage of high technology computerized environments within which watershed models have been developed. These include:

(i) Spatial software. Estimation and representation of hydrologic parameters is based on using the advanced technology of spatial analysis tools such as remote sensing, GIS and GPS in large scale areas. Remote sensing facilitates the development of a spatial and temporal database for modelling a watershed based on representation of parameters in GIS. Physical geographers develop models based on a geomorphologic point of view of a watershed and erosion processes. Dermo at al. (1994) developed the LImburg Soil Erosion Model (LITEM) which is based on spatial variability of runoff and soil erosion. This model is one of the first examples of a physically-based hydrologic and soil erosion model in a raster GIS environment.

(ii) Artificial intelligence. This is a new alternative for representing the logical relationship between qualitative and quantitative parameters in a watershed in the present and in the future.

1.2 Computational Intelligence Models

Computational intelligence and soft computing approaches exploit a tolerance for imprecision, uncertainty and partial truth to achieve tractability, robustness and low solution cost (Zadeh 1992). These types of approach typically borrow and adapt models from nature such as biological neural networks, evolutionary adaptation, models of human cognition, etc. The field therefore interfaces in many and interesting ways with traditional biological research.

Computational intelligence may be defined as, a computational system that can gather and store information and process it in explicit and implicit manners with a degree of randomness to efficiently solve the problem at hand.

In recent years, fuzzy logic applications have entered the mainstream of academic and industrial research and development to the extent that commercial applications are now common (Kaufmann 1975, Lee 1990a,b, Zimmermann 1994, Cox 1992, Schwartz et al. 1994, Yager and Filev 1994) Fuzziness can be found in many areas including engineering (Schwatz 1994) and in manufacturing (Mamdani and Assilian 1981). There has also been significant progress in efforts to combine fuzzy computation paradigms with other approaches such as neural networks, to create hybrid systems with enhanced capabilities (Kosko 1992, Berenji 1992, Nauck and Kruse 1992). Both neural networks (NN) and fuzzy logic systems

(FLS) deal with important aspects of knowledge representation, inferencing, and learning processes. However, they use different approaches and have their own strengths and weaknesses. Neural Networks (NN) can learn from sample data automatically but lack an explanation ability, whereas, fuzzy logic systems (FLS) are capable of performing approximate reasoning but are usually not self-adaptive. The real power of artificial intelligence lies in the integration of NN and FLS. Existing integration methods can be classified into three broad categories: (i) building FLS with NN, (ii) converting NN into FLS, and (iii) combining FLS and NN into a hybrid system. A variety of applications involving the integration of NN and FLS are reviewed and the direction for further research in this area is suggested.

Many scientists and engineers now use the paradigms of fuzzy computation to tackle problems that are either intractable or unrealistically time-consuming to solve through traditional computational strategies.

The paradigms of evolutionary computation such as genetic algorithms (GA) (Kim and Kim 1995, Negoita and Roventa 1994, Perneel et al. 1995, Zhang et al. 1995, and Castro 1995), parallel algorithms, evolution strategies, genetic programming, classifier systems, and combinations or hybrids thereof have been applied to problems that fall under the categories described by Lee (1990a).

Recently there have been vigorous initiatives to promote cross-fertilization between the hydrological and agricultural sciences, expert systems and soft computing paradigms (Bardossy and Duckstein 1995, Ferson and Kuhn 1992) and also to combine these paradigms with other approaches such as neural networks to create hybrid systems with enhanced capabilities. Shen and Leitch (1993) used qualitative reasoning and fuzzy set theory to produce an effective algorithm for simulation of dynamic systems by qualitative data. Zeng and Singh (1995) developed an approximation MIMO model to demonstrate the approximation properties such as fuzzy approximation mechanism, global approximation bound, convergence property, and approximation capability of fuzzy systems. Turksen and Tian (1995) investigated a rule search for a fuzzy expert system based on the overall similarity measure (OSM) for industrial application.

1.3 Research Objectives

The overall objective of this study was to apply artificial intelligence and soft computing to build a fuzzy expert system to provide hydrologic design information for a watershed, and to predict the curve number (CN).

Specific objectives were:

(i) To derive relationships among hydrologically significant watershed components of soil, cover, land use, moisture content and impermeability based on logical relations.

(ii) To modify the discrete SCS method of classifying hydrologic soil groups to facilitate modeling the real continuity of soil and moisture conditions.

(iii) To enable prediction of the hydrologic curve number (CN) using computational artificial intelligence, and

(iv) To develop a PC-based fuzzy expert system for use by soil and water managers to determine the CN for runoff potential as used by many hydrology models.

2. LITERATURE REVIEW

2.1 Overview of Current Conventional Models

The concept of modelling is generally based on a mathematical method in the form of an analytical or numerical technique, used to represent the physical properties of the system. The mathematical problem is then solved by an acceptable method and the result should then represent the response reality of the physical phenomena modelled. Ferreira and Smith (1988) presented the basic concepts of hydrologic and watershed modelling under the following categories:

(i) Problem definition; identification of the hydrologic quantities of interest to the intended application.

(ii) System conceptualization; design of algorithms and conceptualization of the modelling processes within the prototype system that need to be modelled in order to estimate the desired quantities.

(iii) Mathematical formulation; this part of a model is based on previous study and some selection of formulation to represent each identified component of the hydrological processes.

(iv) Re-organization and synthesization; the selected process equations are transformed into a computational framework to reach the desired responses for the system as a whole. This stage commonly involves computer programming.

(v) Model validation; the model has to be tested in two stages. The first stage comprises the verification of the logical functioning of the parameters by testing the computer program, while the second stage is to validate the adequacy of the model by application to a real situation.

Current mathematical models of watershed hydrology can be classified in terms of quantities and qualities of the variables used for prediction, spatial scale, time scale, generality of the model and the applicability of the model. One of the simplest models (a single equation) that explain the relationships among components in a watershed is the Universal Soil Loss Equation (USLE) (Wischmeier 1959). A recent update of the USLE has been the development of the Revised Universal Soil Loss Equation (RUSLE) (Renard et al. 1991).

This again is an empirical model supported by a huge number of more recent experimental studies. The RUSLE model, which uses USLE factors, greatly extends the USLE by including runoff, interrill and rill erosion, sediment size and densities, erosion of channels, and deposition of detached soil in parts of the watershed.

By the early 1950s digital computers become generally available and hydrologists began to explore the application to hydrologic problems (Decoursey and Snyder 1969).

In 1960 Linsley and Crawford reported their work on the development of the Stanford Watershed Model I (SWM). This was a relatively simple model using daily rainfall, a simple infiltration function, and a combination of the unit hydrograph and recession functions to generate the mean daily flow hydrograph.

Development also began on a model later named Areal Non-point Source Watershed Environment Response Simulation (ANSWERS) (Huggins and Monke 1966). This model was initially developed to simulate the response of small (2 ha) agricultural watersheds, during and immediately following a rainfall event. Its primary application was initially for flood generation. However, it was further developed for evaluating strategies for controlling nonpoint source pollution from intensively cropped areas. The ANSWERS model uses a distributed parameter approach compared to a lumped parameter approach used in previous models. A spatially distributed model has the potential for providing a more accurate simulation of natural catchment behaviour, and the ability to simultaneously simulate conditions at all points within the watershed (Beasley and Huggins, 1991). A complete discussion of the development of the ANSWERS model is presented by Huggins and Monke (1966) and Beasley et al. (1980).

The Cool Season Soil Erosion Model (COSSEM) was developed specifically for soil erosion modelling in Prince Edward Island (Burney and Edwards 1992). COSSEM is based on the ANSWERS model and is a major revision of the model presented by Burney (1977). The model is designed for specific application to 'cool season' nonpoint source pollution modelling in Prince Edward Island. The ANSWERS model, which was developed for summer conditions has limited use in regions such as Prince Edward Island where the soil is subjected to many freeze-thaws cycles from November through April, during which the major runoff events occur. COSSEM was written to be used on catchments or terrace systems having channels, as well as non-channelized catchments such as fields or rainfall simulator plots.

A spatially distributed model, Water Resources Management Model (WRMM) was designed for continuous modelling over a full summer period with primary application to Prince Edward Island (Mbajiorgu 1992). WRMM was aimed at modelling the effects, on summer runoff events, of installing soil conservation structures.

The Chemicals, Runoff and Erosion from Agricultural Management Systems (CREAMS) model models the daily response of agricultural fields (Knisel 1980). CREAMS may be viewed as a fundamental model whose component sub-models have been applied in later models. CREAMS consists of three sequential modules: hydrology, erosion, and chemicals (plant nutrients or pesticides). The erosion section of the CREAMS model is much more sophisticated than either the hydrology or chemicals sub-models.

The Groundwater Loading Effects from Agricultural Management Systems (GLEAMS) model was developed from CREAMS (Knisel et al. 1990). GLEAMS includes shallow groundwater (soil profile) effects as well as all the components included in CREAMS. Most of the GLEAMS parameters are the same as CREAMS, including the surface representation.

Following the development of EPIC (Erosion Productivity Impact Calculator), Williams et al. (1985) used many of the concepts in developing the Simulator for Water Resources in Rural Basins (SWRRB) model. SWRRB is a semi-spatial adaptation of the CREAMS model, making it suitable for applications involving large, complex watersheds. The major processes included in the model are surface runoff, percolation, return flow, evapotranspiration, channel transmission losses, pond and reservoir storage, erosion and sedimentation, and crop growth. Arnold et al. (1991) added water quality to the SWRRB model and named the model Simulator for Water Resources in Rural Basins-Water Quality (SWRRBWQ). The objectives of these models are to predict the effect of management decisions on water, sediment, nutrient, and pesticide yields with reasonable accuracy for ungaged rural basins. Each of the CREAMS, GLEAMS, SWRRB and SWRRWWQ models uses the Soil Conservation Service (SCS) Curve Number (CN) technique for predicting surface runoff depth on a daily basis.

Both the SWRRB (Arnold et al. 1990) and SWRRBWQ models have size limitations (up to a few hundred square kilometres) and numbers of such watersheds (maximum ten subbasins). The SWRRB model also had a simplistic routing structure with outputs transferred from the sub-basin outlets directly to the basin outlet. To improve modelling simulation, a model called ROTO (Routing Outputs to Outlet)(Arnold et al. 1995) was developed to take the outputs from multiple SWRRB runs and route the flows through channels and reservoirs. However, this combination required considerable computer storage, and manual intervention.

Arnold et al. (1995) developed a more sophisticated model called SWAT (Soil and Water Assessment Tools) to overcome these limitations. Thus, the SWAT model was developed by merging SWRRB and ROTO into one basin scale model. SWAT allows a basin to be divided into hundreds or thousands of grid cells or sub-watersheds. SWAT is a continuous time model (daily step) that facilitates simulation of long-term impacts of management (e.g., reservoir sedimentation over 50-100 years) and timing of agricultural practices within a year (e.g., crop rotations, planting and harvest dates, irrigation, fertilizer, and pesticide application rates and timing).

In 1987, the United States Department of Agriculture (USDA) Agricultural Research Service (ARS) began developing a new process-based technology to replace the empirical USLE technology as the primary means for predicting soil erosion (Foster 1987). The resulting model is known as the Water Erosion Prediction Project (WEPP), and is in the form of a user-friendly computer program. The WEPP model separates water erosion into rill and interrill components, and combines the results to calculate total erosion on a storm-by-storm basis over an entire growing season. WEPP allows application to a diversity of environmental conditions, as well as improvements in simulating the effects of soil moisture, topography, residue cover, and seasonal changes in crop growth (Elliot et al. 1989). WEPP can be used in either a continuous simulation mode or a single event mode.

2.2 SCS Methods of Direct Runoff Prediction

The Soil Conservation Service (SCS) method was developed by SCS hydrologists for estimating direct runoff from storm rainfall (Mockus 1964). The principles of the method are not new, but they are continually put to new uses. The SCS method builds a bridge between meteorological and physiographic parameters in that the volume and rate of runoff depends on both meteorologic and watershed characteristics, and estimation of runoff utilizes an index to link these two factors. The precipitation volume is the single most important meteorological characteristic used in estimating the volume of runoff. This meteorological parameter is influenced by physiographic variables such as elevation, local slope degree, orientation of the slope, and the moisture source. The soil type, land use, and hydrologic condition of the cover are the watershed factors that have the most significant impact in estimating the volume of runoff. The antecedent soil moisture is also an important determinant of runoff volume.

The Soil Conservation Service developed several empirical equations for various soil and land use practices. The principal application of the method is in estimating quantities of runoff in flood hydrographs or in relation to flood peak rates. Four types of runoff can be considered as:

(i) Channel runoff: occurs when rain falls on a flowing stream. It contributes directly to the flood hydrograph. However, its magnitude is negligible.

(ii) Surface runoff: occurs only when the rainfall rate is greater then the soil infiltration capacity. This type of runoff occurs after the initial demands of interception, high infiltration capacity, and surface storage have been satisfied.

(iii) Interflow: occurs when infiltrated rainfall percolating downwards encounters a soil layer of low transmission. Flow then travels above the layer and may reemerge on the soil surface downhill as a seep or spring.

(iv) Baseflow: occurs as a fairly steady flow from natural deep storage below the vadose zone. This type of flow seldom appears soon enough after a storm to have any influence on the peak rate of runoff for that storm, but baseflow from a previous storm will increase the flow rates.

In arid regions the flow on smaller watersheds is nearly always surface runoff, but in humid regions it is generally more of the subsurface type. The SCS method lumps all types of runoff into direct runoff, which consists of channel runoff, surface runoff, and sub-surface flow in unknown proportions. The Curve Number (CN), is an indicator of the probability of flow type proportions: the larger the CN the more likely that the estimate is of surface runoff. The value of the curve number (CN) ranges from 0 to 100 and depends on the watershed properties such as soil type, cover type, treatment, antecedent soil moisture condition (AMC), and hydrologic condition. McCuen (1982) and Boughton (1989) argued that the main source of error in estimating runoff by SCS Curve Number is the selection of the CN.

The best fitting relationship for estimating runoff, found by empirical analysis was:

$$Q = \frac{(P - 0.2S)^2}{P + 0.8S} \qquad (2.1)$$

where P is rainfall (mm), Q is runoff (mm) and S is the maximum potential difference (storage) between rainfall and runoff (mm).

The SCS method uses the runoff Curve Number CN, which is related to storage by:

$$S = (\frac{25400}{CN} - 254) \qquad (2.2)$$

Many models including CREAMS (Knisel 1980), GLEAMS (Leonard et al. 1987), AGNPS (Young et al. 1994a, b), SEDIMOT (Wilson et al. 1986), SPUR (Wight, 1983), SWRRB (Williams et al. 1985), SWMM (Huber et al. 1981), SWAT(Arnold et al. 1995) and a sophisticated graphical user-friendly interface commercial model, Watershed Management System (WMS v.5) (Charistiansent 1997) all use the CN method for runoff prediction. All are based on the work originally done by Mockus (1964).

The Soil Conservation Service (SCS) Curve Number (CN) procedure requires an initial organization of the physiographic data base for the problem watershed. The SCS method is composed of the three principal dimensions of soil (hydrologic soil groups), runoff (volume, peak flow, lag, time of peak), and cover-management (vegetation, management practices) (McCuen et al. 1984).

2.2.1 Hydrologic Soil Group Determination

Soil properties influence the process of generation of runoff from rainfall and they must be considered, even if only indirectly, in methods of runoff estimation. In agricultural land, the minimum rate of infiltration is obtained for a bare soil after prolonged wetting. In the SCS method the soil properties are represented by a hydrologic parameter, which indicates the runoff potential of the soil and is the qualitative basis of the classification of all soils into four groups. The classification is broad but the groups can be divided into subgroups. Schmidt and Schulze and Schmidt (1987a,b) adapted the SCS method by changing the four basic soil hydrology groups for Southern African soils. Three intermediate soil groups have been used in the classification of soil forms and series. These groups are A/B, B/C, and C/D, giving a total of seven soil groups.

The definition of soil groups is based on soil texture, infiltration rate, transmission rate, and runoff potential. The hydrologic soil groups, as defined by SCS, consist of four groups which are identified by the letters A, B, C and D. Soil characteristics that are associated with each group are as follows:

Group **A**: Soils with very low runoff potential, having high infiltration rates even when thoroughly wetted and consisting chiefly of deep sand, deep loess and gravelly sand. These soils have a high rate of water transmission.

Group **B**: Soils having moderate infiltration rates, comprising shallow loess and sandy loam (with fine to moderately coarse textures). These soils have a moderate rate of water transmission.

Group **C**: Soils having slow infiltration rates when thoroughly wetted (high runoff potential). The soil textures comprise chiefly clay loam and shallow sandy loam. These soils have a moderate rate of water transmission.

Group **D**: Soils having very low infiltration (very high runoff potential). Soils consist chiefly of clay soils , soils with very poor drainage or permanent high water table and soils with a claypan or clay layer near the surface. These soils have a very slow rate of water transmission.

The SCS soil group can be identified at a site by one of three methods:

(i) Soil characteristics,

(ii) Soil survey classification (county soil map), or

(iii) Minimum infiltration rate.

The soil characteristics associated with each group are listed above. County soil surveys give a detailed description of the soils at locations within a county. These surveys are usually the best means of identifying the soil group. Soil analysis can be used to estimate the minimum infiltration rates, or best estimation of some soils are by measurement of infiltration rates in situ (e.g., using a double ring infiltrometer), which can be used to classify the soil using the values in Table 2.1.

Table 2.1. Minimum infiltration rates for hydrologic soil groups (SCS 1972).

Soil Group	Minimum infiltration rate (mm/hr)	Class
A	7.5 - 12.5	High
B	3.75 - 7.5	Moderate
C	1.25 - 3.75	Slow
D	0 - 1.25	Very Slow

Classification according to texture is presented in Table 2.2. The relationship between soil classes based on the SCS soil classification and % sand, %clay and minimum infiltration rate are presented in Figs 2.1 and 2.2, respectively.

Table 2.2. Soil texture and permeability classification (after: Cook et al. 1985).

Texture	Permeability class	Hydrologic soil group
Silty clay, clay, heavy clay	6	D
Silty clay loam, sandy loam	5	C-D
Sandy clay loam, clay loam	4	C
Loam, silt loam, silt	3	B
Loamy sand, sandy loam	2	A
Sand	1	A

2.2.2 Hydrologic Condition (Cover)

Hydrologic condition is a qualitative definition of cover density. This condition is classified as poor, fair, or good as defined below:

(i) Poor condition: an area that is grazed heavily and has mulch or plant cover on less than 50% of the area.

(ii) Fair Condition: an area not heavily grazed and with cover ranging from 50% to 75% of the area.

(iii) Good condition: an area lightly grazed and has plant cover on more than 75% of the area.

These classifications of poor, fair and good are fuzzy, but discrete, and can be translated to statements in fuzzy set theory.

Wood and Blackburn (1984) evaluated the hydrologic soil group on rangelands by using the SCS runoff method. As a result of these studies, they recommended that the hydrologic soil groups should be abandoned, or greatly modified, for use in estimating infiltration rates and runoff in arid and semi-arid rangeland, and criteria should be developed which make use of surface soil characteristics.

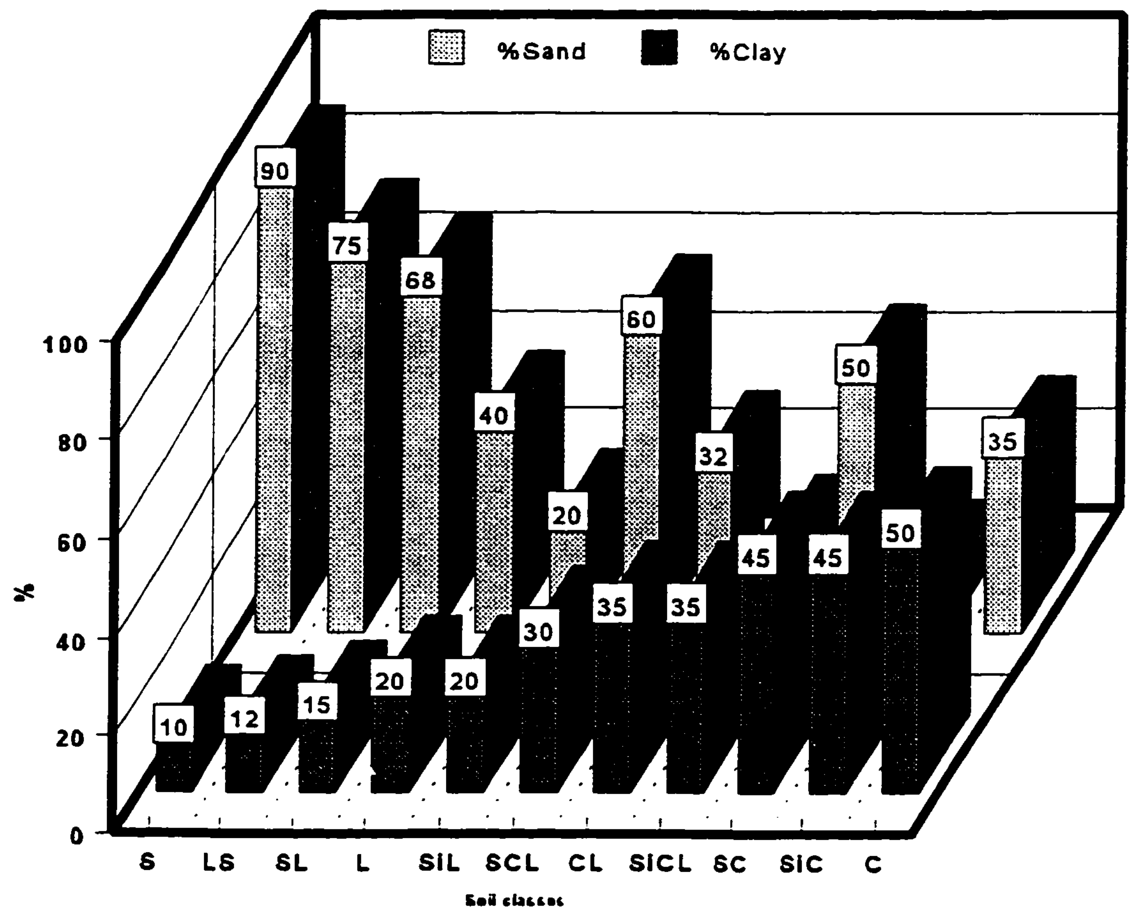


Fig 2.1 Relationship between soil classes based on the SCS method of soil classification with % sand and % clay (after: Schueler 1987).

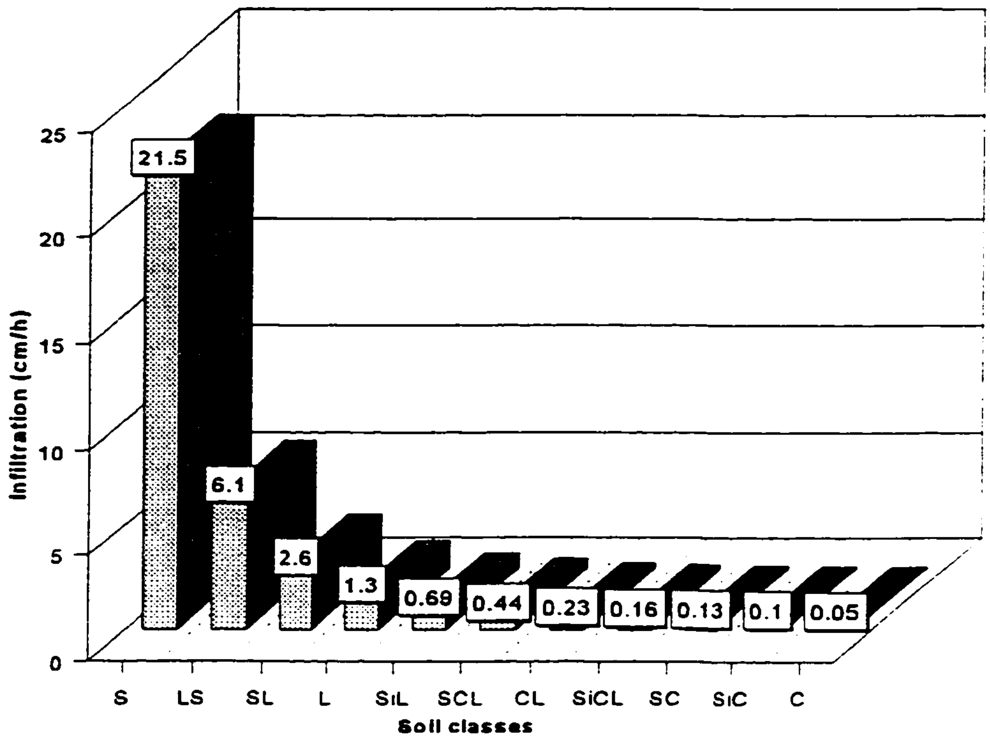


Fig 2.2 Relationship between soil classes based on the SCS method of soil classification and infiltration (after: Schueler 1987).

2.2.3 Surface Treatment Effects

Land treatment applies mainly to agricultural land uses and it includes mechanical conservation practices such as planting on the contour as against up-and-downhill planting and conservation tillage. These activities decrease the runoff potential. In an urbanizing setting, as the population increases, the surface land is developed for urban use and a region is transformed from the natural state to a totally manmade state. New structures, which add large amounts of impervious areas to the watershed, decrease the water storage (infiltration capacity approaches zero).

2.2.4 Antecedent Moisture Conditions (AMC)

The effects of antecedent moisture condition (AMC) were originally presented by SCS (1972) as a discrete value for three specific conditions of dry, average and wet. These AMC classes are estimated from five-day antecedent rainfall for either the dormant season or the growing season, as applicable.

2.2.4.1 Determination of CN for AMC I and AMC III

The CN depends on soil type, cover type, treatment, hydrologic condition, and antecedent moisture condition (AMC). SCS (1972, 1986) presented CN based on AMC II (average). One of the common error-causing components in the CN estimation procedure is the AMC. Tables for converting curve numbers from average (AMC II) to wet (AMC III) or dry (AMC I) conditions also can be found in these publications. Yoon and Padmanbhan (1995) estimated AMC I and AMC III by using a non-linear procedure to obtain the regression relationships. Their fitted polynomial equations are given below.

(i) For conversion from AMC II to AMC I:

$$CN_{AMC\,I} = -.24236 + 0.4878(CN_{AMC\,II}) - 7.4104E{-}3(CN_{AMC\,II})^2 + 3.9971E{-}4 \\ (CN_{AMC\,II})^3 - 5.249E{-}6(CN_{AMC\,II})^4 + 2.5014E{-}8(CN_{AMC\,II})^5 \qquad (2.3)$$

(ii) For conversion from AMC II to AMC III:

$$CN_{AMC\ III} = 0.61611 + 2.4914(CN_{AMC\ II}) - 4.4208E-2(CN_{AMC\ II})^2 + 6.6839E-4 (CN_{AMC\ II})^3 - 5.4912E-6(CN_{AMC\ II})^4 + 21.7112E-8(CN_{AMC\ II})^5 \tag{2.4}$$

2.2.5 Runoff Characteristics

The volume and rate of runoff depends on both meteorologic and watershed characteristics, and estimation of runoff requires a method to represent these two factors. The rainfall volume is the single most important meteorological characteristic in estimating the volume of runoff. The soil type, land use, hydrologic condition of the cover and antecedent moisture condition are the watershed factors that will have the most significant impact in estimating the volume of runoff.

The SCS method has been applied by many hydrologists(Hawkins, 1978) for direct runoff estimation. Jackson and Rawls (1981) applied the SCS curve number method to an urban area using Landsat data. They concluded that Landsat data bases can be used in the CN estimation process. The CN method also has been suggested as an appropriate method for infiltration estimation (Chen 1982, Aron et al. 1977, Kumar and Jain 1982).

Curve numbers range from 0 to 100 depending on the watershed properties such as soil type, cover type, treatment, antecedent soil moisture conditions (AMC), and hydrologic condition. Typical values of CN are given in publications based on AMC II (average). This average does not correspond to the actual watershed condition and CN can be adjusted appropriately by use of expert systems.

Hawkins et al. (1985) suggested a probabilistic interpretation of AMC in the curve number (CN) method, and applied the results to the problem of determining CN from rainfall-runoff data. Any P and Q pair with Q>0, leads to an S, and to a CN using the relationship:

$$S = 5(P + 2Q - \sqrt{(4Q^2 + 5PQ)}) \tag{2.5}$$

Bullock et al. (1990) suggested a modification of CN to account for varying moisture content with depth in the soil profile.

The original SCS infiltration model was developed for predicting total runoff volumes from watersheds where only total precipitation data were available. Kumar and Jain (1982) extended the SCS model to determining flood hydrographs, in which a plot of accumulated runoff versus accumulated rainfall shows that runoff starts after an initial abstraction. They showed that the infiltration rate depends on the rate of rainfall and, CN as representative of soil hydrologic groups, cover complex, and land use.

2.3 Artificial Intelligence and Combined Models

2.3.1. General

The concept of expert systems (ES) and computational intelligence originated from research in the area of artificial intelligence (AI). This knowledge is a branch of computer science and electrical engineering with a soft computing point of view, and is concerned with the development of systems that exhibit human-like intelligence from logic orders phenomena.

Research in this area began in the 1950's and the target point was the general process of human reasoning (Minsky 1986, Newell et al. 1990). Further research indicated the general patterns of reasoning a person uses for any problem (Feigenbaum 1977). In developing an expert system an attempt is made to represent the knowledge of an expert in a specific area of application. In general, a specific expert system then assists a user in making a decision, or solving a problem, within a specific domain of expertise by mimicking an expert. Jackson (1986), Barr and Pratt (1988) identified the characteristics of an expert system such as: complexity, that normally needs a considerable amount of human expertise; high performance in terms of speed and reliability; and the justification of solutions and recommendations to the user interface. An expert system has three main components: the inference engine that drives the decision making process, a knowledge base that organizes known facts and information necessary in making a decision, and the user interface.

Many of the early efforts at expert system construction were based on the collection of large numbers of rules that captured empirical associations about their domain. A program such as MYCIN (Shortliffe 1976) is composed of several hundred rules that encode the experience of human expertise. Essentially, the area of expert systems is the world of uncertainty. For instance, in the field of soil and water engineering, we can find several qualitative expressions such as Slow, Medium, High, Good, Very Good, Rapid, and so on. These expressions are kinds of classes that only can be defined in an expert system domain as quantitative values. There are three common knowledge representations and inference process schemes, which are Rules, Semantic Networks and Frames.

2.3.2 Current Use of Expert Systems in Hydrology

The development and application of Artificial Intelligence (AI) in agriculture as well as in hydrology is a new domain that has emerged only in the last five to ten years. In this time there have been many efforts to develop expert system applications, but few systems are operational. Examples are Water Efficient Landscape Planner(WELP) developed by Adams et al. (1992) to reduce water use in residential areas, and Decision Support System (DSS) for risk assessment of groundwater quality developed by Embleton and Engel (1992). Site layout, water well condition, septic systems and fertilizer and pesticides applications are all assessed by DSS. Rouhani and Kangari (1987) developed an expert systems for landfill site selection with application to urban areas. Newell et al. (1990) applied a graphical support system to groundwater contamination problems. The Snowmelt Runoff Model (SRM) was originally developed by Martinec (1975) for simulation of snowmelt runoff for small European basins. Engman et al. (1986) developed an expert system by incorporating the SRM model into a model called EXSRM. SRM was written in FORTRAN and the source code for the SRM model was converted in ZETA LISP with a FORTRAN tool kit. The EXSRM model appears to be a very effective means for improving the utility of complex simulation models. Recently, Varas and Von Chrismar (1995) developed an expert system for selection of methods to calculate design flood flows. This expert system is able to select the appropriate method based on the size of the watershed.

Schroeder et al. (1995) developed an expert system for irrigated wheat crop management with an integrated architecture of the knowledge based system.

Arbour (1992) developed an expert system for soil erosion control, planning and soil conservation recommendations in Prince Edward Island (PEI). The expert system is based on field physiographic information, human expert logical relations in practical agricultural problems and the use of the USLE to estimate soil loss. This expert system demonstrated a strong potential for developing soil conservation recommendations. The knowledge base was structured in a frame formulation.

Yoon and Padmanabhan (1995) developed an expert system called Expert Hydrologist for synthetic rainfall and direct runoff prediction for 12 mid-western states of the USA. Expert Hydrologist is a mixed model with some external programming, obtained by implementing OBJECT LEVEL 5 as an expert development shell. The main problem with using a very sophisticated expert shell is then the dependency of the expert system hydrology on the expert shell.

DaRe (1993) developed an expert system for groundwater protection (well protection) called PROTEX. This expert system evaluates the contaminant transport around a well based on the physical and chemical properties of the soil and water.

For real-time hydrologic control application, the artificial intelligence process often has superiority over conventional methods. Shukla et al. (1996) used artificial neural networks (ANN) to solve the Boussinesq Equation for transient drainage design. They indicated that the ANN was about 2,600 times faster than the procedure of Numerical Method of Lines (NMOL) developed by Skaggs (1976).

Fuzzy logic based AI has a vast domain of categories of application such as fuzzy clustering (Bezdek and Hathaway 1992), fuzzy optimization (Buckley and Hayashi 1993), fuzzy relational equations (Giuclea et al. 1996), fuzzy expert systems (Kim and Kim 1995), fuzzy classifier systems (Bonarini 1993), fuzzy information retrieval and database queering (Kraft et al. 1995), fuzzy decision making, financial and economic models (Cox 1993),fuzzy regression analysis (Ghoshray 1996), and fuzzy pattern recognition and image processing (Albert et al. 1990).

In environmental engineering applications such as hydrology, agriculture, food processing, forestry, and waste management the application of fuzzy logic becomes an attractive area of study. Duckstein et al. (1990) applied fuzzy set theory for optimization of water resources.

Guillermo and William (1989) investigated forest planning with respect to decision-making under fuzzy environments. David (1989) applied fuzzy graphs to analyze ecological modeling of forest land. Woldt et al. (1992) proposed the use of a geostatistical method such as kriging, together with fuzzy set theory for mapping groundwater in three-dimensions. Bardossy et al. (1990) used fuzzy linear regression in hydrology for soil electrical resistivity and soil permeability. Bardossy and Disse (1993) used a fuzzy rule-based model for infiltration. Bardossy et al. (1995) applied fuzzy logic to study the movement of water in one, two and three-dimensions. Smith and Eli (1995) developed a neural network model to predict the peak discharge and time of peak from rainfall patterns, and Raman and Sunilkumar (1995) investigated artificial neural networks in the field of synthetic-inflow for multivariate time series in a watershed. Davidson (1997) applied fuzzy logic control in food processing.

McBratney and Moore (1985) implemented fuzzy set concepts for climatic classification of forest and agricultural areas. Burrough (1989) applied a fuzzy mathematical approach for soil survey and land evaluation. Flick (1993) used fuzzy set theory in modeling management of agricultural ecosystems. Hansen (1997) developed a fuzzy expert system called SIGMAR for marine forecasting based on a climatological analysis system and operational meteorology (Murtha, 1995). Ameskamp (1997) developed a three-dimensional continuous soil-landscape rules based fuzzy system for modelling soil information with a continuous perspective of the soil called the TRCS model. This model is integrated in the GIS GRASS system to explicitly produce a soil-landscape model with a low cost of operation.

3. COMPUTATIONAL INTELLIGENCE MODEL

The major terminology and some basic definitions are introduced in this chapter which provides an overview of the application methodology. Specific relevant applications are presented in the next chapter.

3.1 Fuzzy Set Theory

Fuzzy set theory provides the mathematical foundation for the description and handling of human's imprecise knowledge. The theory of fuzzy sets was originally proposed by Zadeh (1965) as a theory of graded concepts in which everything is a matter of degree. Fuzzy set theory is a super set of conventional (Boolean) logic that has been extended to handle the concept of partial truth - truth values between "completely true" and "completely false".

The concepts of fuzzy set theory, as a means for expressing the degree of ambiguity in human thinking and quantifying it in real numbers, allows uncertain phenomena to be treated mathematically. This theory is sometimes called *modeling of uncertainties*, as the following kinds of uncertainties are distinguished:

(i) Statistical uncertainty. An event occurs with a given probability as, for example, the result of the throw of a dice,

(ii) Lexical or linguistic uncertainty. The imprecise description of an object like large apartment, low price, rapid, etc.,

(iii) Informational uncertainty. The uncertainty caused by missing or incomplete information such as credit worthiness.

By definition if X is a collection of objects each denoted generally by x, then a fuzzy set $\hat{A}$ in X is a set of ordered pairs, such that,

$$\hat{A} = [(x \, , \, \mu_{\hat{A}} \, (x)) \mid x \, \varepsilon \, X] \tag{3.1}$$

Where $\mu_{\hat{A}}(x)$ is called a membership function, degree of compatibility, or degree of truth of x in Â, which maps the set X to the membership space M.

The membership function can be any real-valued function but, in general, normalized membership functions with values between 0 and 1 are used. There, therefore, seems to be a similarity to the description of possibilities which does not occur in reality (Zadeh and Kacprzyk 1992). The extension of binary logic towards an interval enables grey values. Instead of only black or white (definitely belongs to or does not belong to) grey levels are introduced. Therefore, an element can partially belong to a set. Because of this an early fault detection, based on trends and a smooth starting of control actions, is possible before a crisp limit is exceeded.

3.1.1 Fuzzy Operators

Fuzzy logic offers an approach towards quantifying human decision making by handling imprecise knowledge. This offers the possibility of moving the man-machine interface of technical systems towards the human operator. Some of the mathematical operators in fuzzy set theory are presented below.

3.1.1.1 Intersection

The intersection of two fuzzy sets A and B can be calculated by different mathematical operations such as minimum, bounded difference, and algebraic product. For example:

$$
\begin{aligned}
\textit{Minimum}: \quad & \mu_{S1} = \min(\mu_a\ ,\ \mu_b) \\
& \mu_{S2} = \min\ (\mu_a, \mu_b\ )\ \textit{if}\ \max\ (\mu_a, \mu_b\ ) = 1 \\
& \qquad\ \ 0 \quad \textit{else} \\
\textit{Bounded difference}: \quad & \mu_{S3} = \max\ (0,\ \mu_a + \mu_b - 1) \\
& \mu_{S4} = \frac{(\mu_a \cdot \mu_b\ )}{2 - (\mu_a + \mu_b - \mu_a \cdot \mu_b\ )} \\
\textit{Algebraic product}: \quad & \mu_{S5} = \mu_a \cdot \mu_b \\
& \mu_{S6} = \frac{(\mu_a \cdot \mu_b\ )}{(\ \mu_a + \mu_b - \mu_a \cdot \mu_b\ )}
\end{aligned}
\tag{3.2}
$$

Besides these fixed operators parameterized operators such as those of Dubois, Hamacher and Yager are presented by Zimmermann (1992).

3.1.1.2 Union

The operators for the union of two fuzzy sets A and B are:

$$
\begin{array}{lll}
\textit{Maximum} & : & \mu_{V1} = \max(\mu_a\,,\ \mu_b) \\
 & & \mu_{V2} = \max\,(\mu_a,\ \mu_b\ )\ \textit{if}\ \min\,(\mu_a,\ \mu_b\ ) = 0 \\
 & & \qquad 1 \quad \textit{else} \\
\textit{Bounded}\ \sum & : & \mu_{V3} = \min\,(1,\ \mu_a + \mu_b\ ) \\
 & & \mu_{V4} = \dfrac{(\mu_a \cdot \mu_b\ )}{1 + (\mu_a \cdot \mu_b\ )} \\
\textit{Algebraic}\ \sum & : & \mu_{V5} = \mu_a + \mu_b - \mu_a \cdot \mu_b \\
 & & \mu_{V6} = \dfrac{(\ \mu_a + \mu_b - 2\,.\ \mu_a \cdot \mu_b\ )}{(\ 1 - \mu_a \cdot \mu_b\ )}
\end{array}
\tag{3.3}
$$

For these calculations the result is greater than or equal to the maximum operation in the equation. The union can also be calculated by parametrized operators such as those of Hamaccher and Yager (Zimmermann 1992).

3.1.1.3 Compensating Operators

The compensating operators return a result greater than the intersection and less than the union operators:

$$
\begin{array}{l}
\mu_{C1} = \dfrac{\mu_a + \mu_b - \mu_a \cdot \mu_b}{1 + \mu_a + \mu_b + - 2\,.\ \mu_a \cdot \mu_b} \\
\mu_{C2} = \dfrac{\max\,(\mu_a\,,\ \mu_b}{1 + (\ \mu_a - \mu_b\ )} \\
\mu_{C3} = \dfrac{\mu_a + \mu_b}{2} \\
\mu_{C4} = \dfrac{\min\,(\mu_a\,,\ \mu_b\ )}{1 - (\ \mu_a - \mu_b\ )} \\
\mu_{C5} = \dfrac{\mu_a \cdot \mu_b}{1 - \mu_a - \mu_b + 2\,.\ \mu_a \cdot \mu_b}
\end{array}
\tag{3.4}
$$

Besides these conventional compensating operators, parameterized compensating operators are also used in order to fit an artificial linguistic description according to a given input-output behavior. Operation of implementation of two fuzzy set A and B are presented in Fig 3.1.

3.1.1.4 Complement or Negation

The complement of a normalized membership function is evaluated by calculating:

$$\mu_{\complement \tilde{A}}(x) = 1 - \mu_{\tilde{A}}(x), \quad x \in X \tag{3.5}$$

3.1.1.5 Criteria for Operators Selection

The variety of operators for the aggregation of fuzzy sets make it difficult to decide which one to use in a specific model or situation. Zimmermann (1992) listed eight important criteria according to which an operator can be selected as the most appropriate operator to be used in an application as follows:

(i) Axiomatic Strength. The less limiting the axioms it satisfies, the better the operator.

(ii) Empirical Fit. It is not only important for an operator to satisfy certain axioms from a mathematical point of view, but the operator must also be an appropriate model of real-system behavior. This can normally be proven by empirical testing.

(iii) Adaptability. An operator is independent of the context and semantic interpretation of an application. However, some operators can be adapted to a specific application based on the scale of the system and parametrization.

(iv) Numerical Efficiency. In practice, this might be quite important, in particular when large problems have to be solved.

(v) Compensation. Some operators such as logical "AND" do not allow for compensation. By using a compensatory operator instead of the AND operator the system can be presented correctly.

(vi) Range of Compensation. In general, the larger the range of compensation the better the compensatory operator.

(vii) Aggregating Behavior. The degree of membership in the aggregated set depends very frequently on the number of sets combined. For example, by using *product operators*

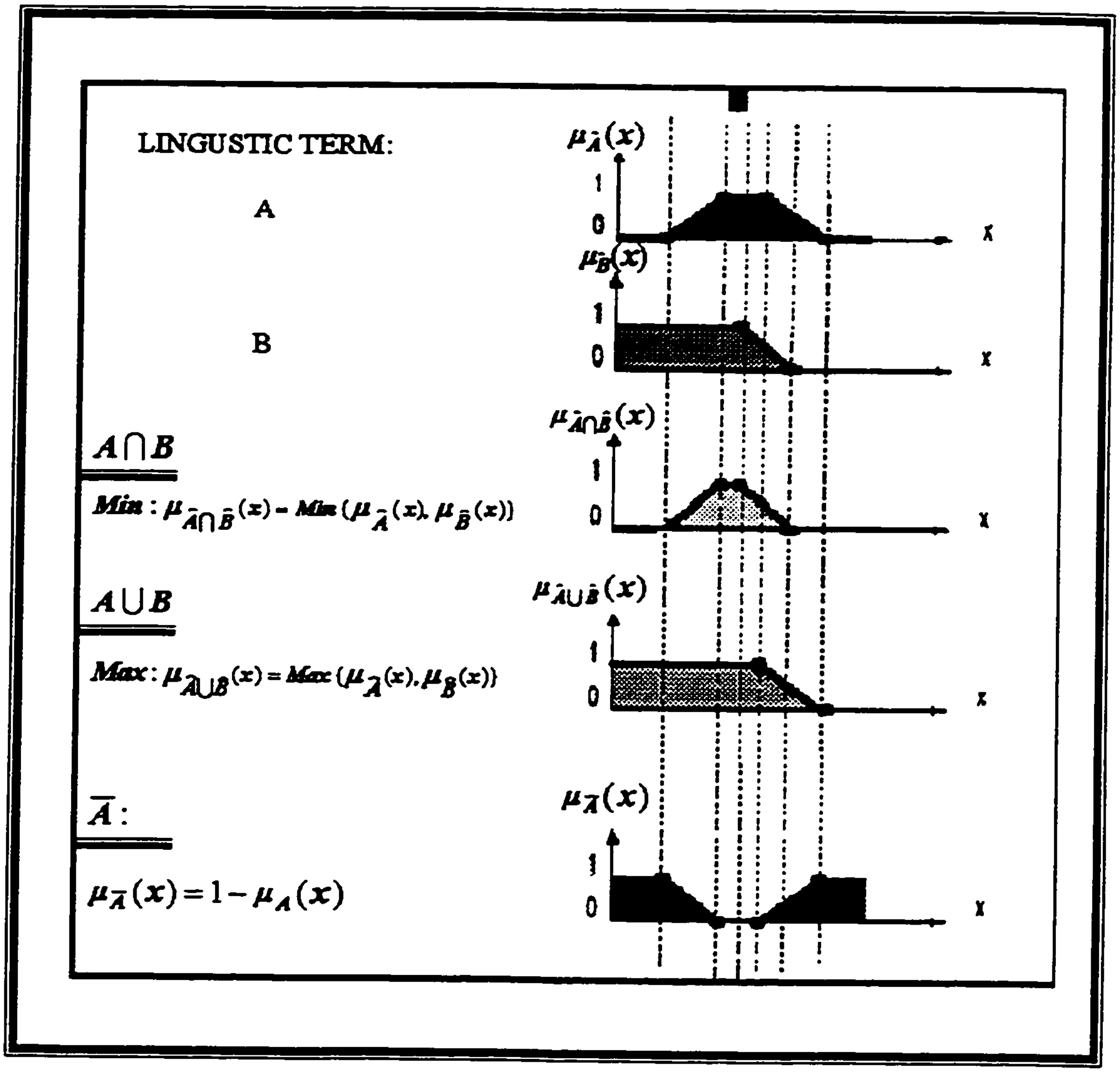


Fig 3.1 Implementation of operation between two membership functions A and B.

each additional fuzzy set will normally decrease the aggregate degrees of membership. This might be a desirable feature (not adequate).

(viii) Required Scale Level of Membership Functions. The scale level (interval, ratio, or absolute) on which membership information can be obtained depends on a number of factors. In general, the operator that requires the lowest scale level is the most preferable from the point of view of information gathering.

3.2 Determination of Fuzzy Memberships

For most control applications the sets that will have to be defined are easily identifiable. For other applications they will have to be determined by knowledge acquisition from an expert or group of experts. Once the names of the fuzzy sets have been established, one must consider their associated membership functions.

In building a fuzzy comprehensive evaluation model, fuzzy memberships are determined and differ for subjective and objective factors (Turksen 1986, 1991, Norwich and Turksen 1984, and Lee 1990a) as discussed below.

3.2.1 Subjective Factors by Numerical Definition

In this case, the grade of membership function of a fuzzy set is represented as a vector of numbers whose dimension depends on the degree of discretization. For a certain membership situation, if u_0 tests are conducted, then the fuzzy membership is calculated by a simple ratio:

$$\mu(x) = \sum \frac{number\ of\ u_o \in A}{total\ number\ of\ u_o} \qquad (3.6)$$

3.2.2 Objective Factors or Functional Definition

To determine fuzzy memberships for objective factors as a function that can express the membership function of a fuzzy set in a functional form, typically a distribution appropriate for the application must be selected. Standard distributions, which are used for objective fuzzy membership calculations, include the *normal distribution*, the *half-decrease distribution*, the *triangular distribution*, and the *trapezoidal distribution*, as shown in Fig 3.2. The fuzzy membership functions associated with these distributions are defined by the mean a and the range b:

For a bell-shape (normal) function (Fig. 3.2a):

$$\mu(x) = \exp\left[-\left(\frac{4(x - a)}{b}\right)^2\right] \tag{3.7}$$

For the half-decrease distribution (Figure 3.2b.):

$$\mu(x) = \begin{cases} 1 & (0 \le x \le a) \\ \frac{1}{2} - \frac{1}{2}\left(\sin\left[x - \frac{a+b}{2}\right] \times \left(\frac{2}{x}\right)\right) & (a < x < b) \\ 0 & (b \le x) \end{cases} \tag{3.8}$$

For the triangular distribution (Figure 3.2c) defined by a triplet (a1, a2, and a3):

$$\mu(x) = \begin{cases} 1 & (x < a1) \\ \left(\frac{x - a1}{a2 - a1}\right) & (a1 \le x \le a3 \\ \left(\frac{a3 - x}{a3 - a2}\right) & (a2 \le x \le a3) \\ 0 & (a3 \le x) \end{cases} \tag{3.9}$$

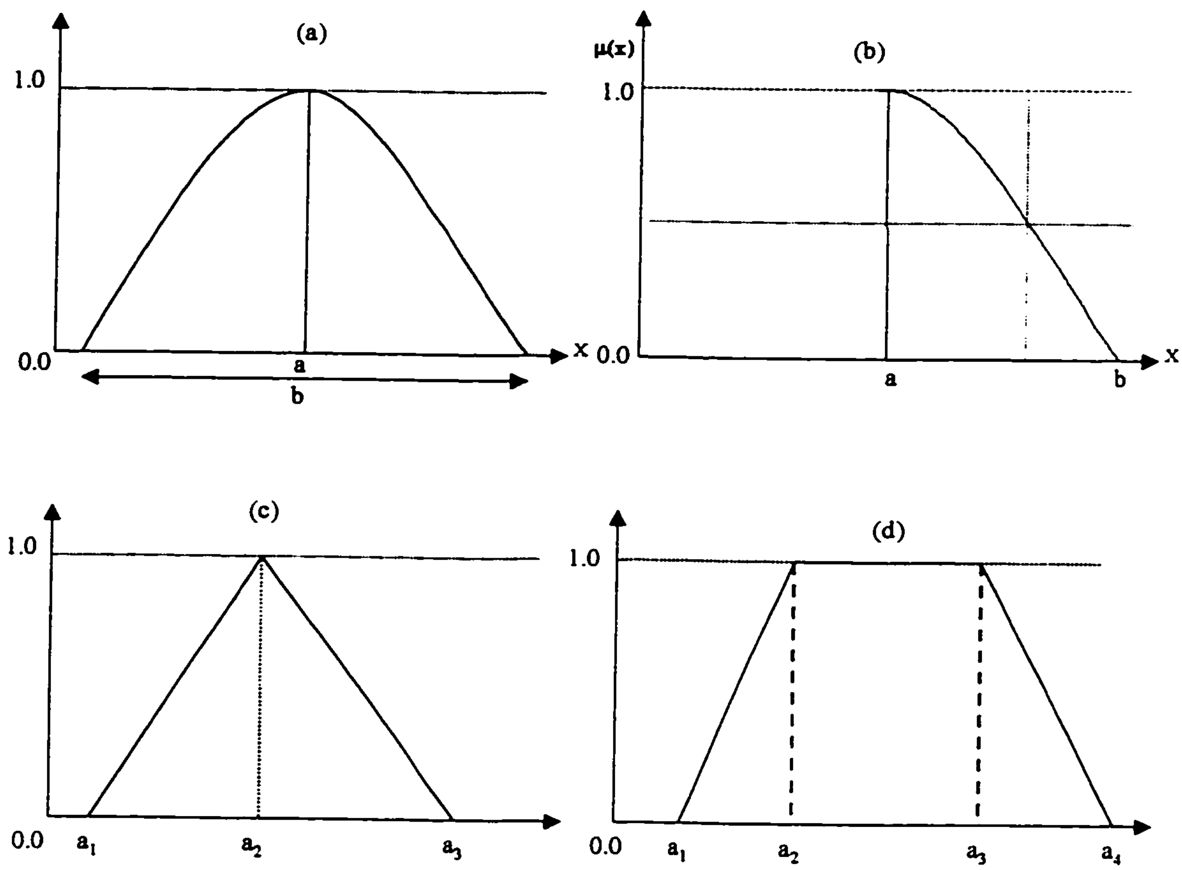


Fig. 3.2 Four types of Fuzzy Membership Distributions: (a) normal; (b) half-decrease; (c) triangular; and (d) trapezoidal.

For the trapezoidal distribution (Figure 3.2d), defined by the quarternion (a1, a2, a3, a4):

$$\mu(x) = \begin{cases} 0 & (x < a1) \\ \left(\dfrac{x - a1}{a2 - a1}\right) & (a1 \le x \le a2) \\ 1 & (a2 \le x \le a3) \\ \left(\dfrac{a4 - x}{a4 - a3}\right) & (a3 \le x \le a4) \\ 0 & (a4 \le x) \end{cases} \tag{3.10}$$

Selection of the type of distribution that best represents the actual situation is based largely on an understanding of the problem and on experience.

3.2.3 Composition of a Fuzzy Relation

All of the fuzzy subsets assigned to each output variable are combined together to form a single fuzzy subset for each output variable. In fuzzy set theory the concept of a fuzzy relation is introduced as a generalization of crisp relations. These fuzzy relations in different product spaces can be combined with each other by an operation called *composition*. There are different versions of composition operators suggested for different applications which differ in their results and also with respect to their mathematical properties. The general form of the *max-* composition* differs in the associative operator *. Two max-* compositions are considered in this section.

As an example, let R_1 (x, y), (x,y)$\in$ (X $\times$ Y) and R_2 (y , z), (y, z) $\in$ (Y $\times$ Z) be two fuzzy relations. The *max-min composition* R_1 max-min R_2 is then the fuzzy set,

$$\mu_{\tilde{R}_1 \circ \tilde{R}_2} = ([(x\ ,\ z),\ \max_y\ (\min\ (\mu_{\tilde{R}_1}\ (x\ ,y),\ \mu_{\tilde{R}_2}\ (y,z))]\ \mid x \in X,\ y \in Y,\ z \in Z) \tag{3.11a}$$

where, $\mu_{\tilde{R}_1 \circ \tilde{R}_2}$ *is the membership function of the fuzzy relation on the fuzzy sets.*

For max-prod Eq. 3.11a can be rewritten as:

$$\mu_{\tilde{R}_1 \otimes \tilde{R}_2} = ([(x\ ,\ z),\ \max_y\ (\mu_{\tilde{R}_1}\ (x\ ,y) \cdot \mu_{\tilde{R}_2}\ (y,z))]\ \mid x \in X,\ y \in Y,\ z \in Z) \tag{3.11b}$$

where, $\mu_{\tilde{R}_1 \otimes \tilde{R}_2}$ *is the membership function of the fuzzy relation on the fuzzy sets.*

In this study the *max-product* or max-avg and *max-min* composition is used. In max composition, the combined output fuzzy subset is constructed by taking the point-wise maximum over all of the fuzzy subsets assigned to a variable by the inference rule (fuzzy logic OR). In SUM composition, the combined output fuzzy subset is constructed by taking the point-wise sum over all of the fuzzy subsets assigned to the output variable by the inference rule. All of the fuzzy relations with max-composition should be done by matrix operation to reach the $\mathbf{R}_1 \circ \mathbf{R}_2$. Eqs. 3.11a and 3.11b for any operation relation matrices $\mathbf{R}_1$ and $\mathbf{R}_2$ may generate different results.

3.3 Fuzzy Logic Control and Fuzzy Expert Systems

Expert systems and fuzzy logic expert systems have one thing in common: both model human experience and human decision-making behavior (both are rule-based systems). An extension of fuzzy theory is a fuzzy expert system or fuzzy logic control (FLC). To date, fuzzy expert systems are the most common use of fuzzy logic. There are, however, also clear differences between expert systems and fuzzy logic control (Zimmermann 1987). These are:

(i) The existing FLC system originated in control engineering rather than in artificial intelligence,

(ii) FLC models are all rule-based systems,

(iii) By contrast to expert systems, FLC serves almost exclusively the control of technological production systems (their domains are even narrower than those of expert systems),

(iv) In general, the rules of fuzzy logic control systems are not extracted from a human expert through the system but are formulated explicitly by the FLC designer, and

(v) FLC inputs are normally observations and their output will be numerical values rather than qualitative expressions.

FLCs are used in several wide-ranging fields, including: linear and nonlinear control, pattern recognition, financial systems, operations research, and data analysis.

By considering the fuzzy system which comprises four principal components: fuzzification, fuzzy rule base evaluation, fuzzy inference engine or decision-making logic and, defuzzification, the flow chart of the process of inferencing of the fuzzy expert system controller (FESC) is shown in Fig 3.3.

3.3.1 Fuzzification

Labeling the crisp value of a numerical input variable with a linguistic term or adjective and determining the corresponding grade of membership is called fuzzification. The fuzzification interface peroms the following functions:

(i) Measures the values of input variables,

(ii) Performs a scale mapping that transfers the range of values of input variables into a corresponding universe of discourse, and

(iii) Performs the function of fuzzification that converts input data into suitable linguistic values which may be viewed as labels of a fuzzy set.

The most commonly used fuzzifier is the singleton fuzzifier and the fuzzy rule base consists of $N = \prod^{n}{}_{j=1} N_j$ rules in the following form:

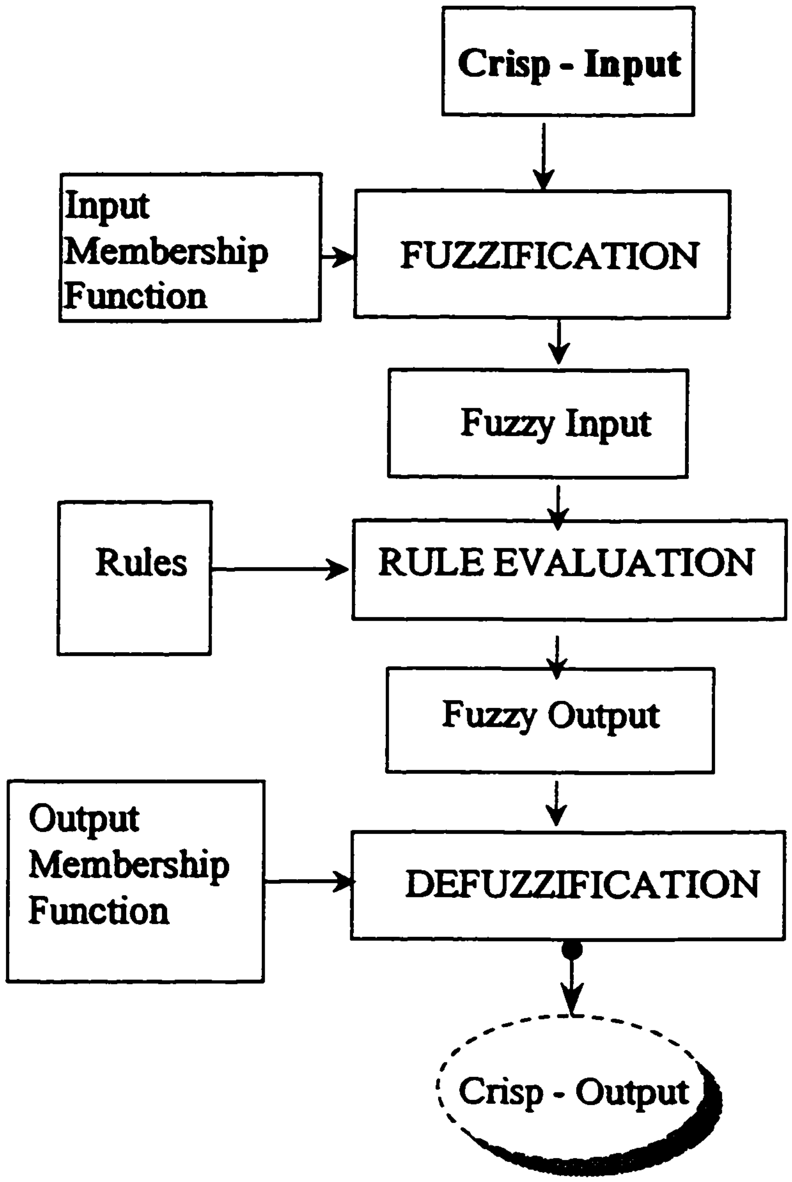


Fig 3.3 Flowchart of the fuzzy expert system controller

$$Ri_1\, i_2 \ldots i_n : \textit{ if } x1 \textit{ is } \mu^1_{i_1} \wedge x2 \textit{ is } \mu^2_{i_2} \wedge \ldots\ldots \wedge x_n \textit{ is } \mu^n_{i_n}$$
$$\textit{then } y \textit{ is the } C_{i1}\, i_2 \ldots i_n \textit{ for } I_1 = 1, 2, \ldots, N_1, i_2, \ldots, N_2, \ldots, \quad (3.12)$$
$$I_n = 1\ ,\ 2, \ldots, N_n$$

where,

x_j ($j = 1, 2, \ldots, n$) are the input variables of the fuzzy system,

y is the output variable of the fuzzy system,

$Ri_1\, i_2 \ldots i_n$ = fuzzy relation by implication in matrix form, and

$\mu^j_{i\,j} \subset U_j$ and $C_{i1\, i2 \ldots in} \subset v$ are linguistic terms characterized by the fuzzy membership function.

3.3.2 Inference Engine

According to the definitions used in artificial intelligence (AI), the determination of conclusions or the generation of hypotheses based on a given input state is called inference.

For operation within the standard control loop this means that the rules define the dependencies between linguistically classified input values and linguistically classified output values. The result is a variable u manipulated according to the input situation. This all occurs in an upper symbolic level first. The implementation can use operators which are partially discussed above.

Common inference strategies are the max-prod inference, which multiplies the whole output membership function, or the max-min inference, which cuts the output's membership function at the top.

The fuzzy inference engine is a decision-making logic module which employs fuzzy rules from the fuzzy rule base to determine a mapping from the fuzzy sets in the input space U to the fuzzy sets in the output space V. If x is an arbitrary fuzzy set in U and $\mu(X)$ is its membership function, then each rule $Ri_1\, i_2 \ldots i_n$ of Eq. 3.12 determines a fuzzy set $V\, \mu_o\, Ri_1\, i_2 \ldots i_n\, i_2 \ldots i_n$ in V based on the sup-star composition (Lee 1990a),

$$V\ \mu_{\circ}\ Ri_1 i_2 \ldots\ldots i_n(y) = \$up_{x\, \varepsilon\, U}\ [\mu(X) * R\ i_1 i_2 \ldots\ldots i_n\ (X\ ,\ y)], \qquad (3.13)$$

where * is assumed to be the algebraic product. This is one of the most commonly used T-norms in applications (Zimmerman 1992). The sup-star composition in the fuzzy inference engine then becomes a sup-product composition and the composition is simplified to,

$$V \ \mu_{\circ} \ Ri_1 i_2 .....i_n(y) \ = \ \$up_{x \varepsilon U} \ [\mu(X) \ \mu i_1 i_2 .....i_n \ (X), Ci_1 i_2 .....i_n(y)]. \tag{3.14}$$

where,

$$\mu_{il \ i_2....i_n} \ (X) \ = \ \mu^1 i \ 1 \ (x_1) \ \mu^2(x_2) \ ......\mu^n \ i_n(x_n). \tag{3.15}$$

The fuzzifer performs a mapping from the sets in V to crisp points in V. However, the symbolic control action cannot be used for a real technical plan as the linguistically obtained manipulated variable has to be defuzzified.

3.3.3 Defuzzification

Defuzzification is the calculation of a crisp numerical value from a space of fuzzy control action. Defuzzification is usually the most time-consuming operation in fuzzy processing.

In most cases several rules will be invoked by different fuzzy terms and therefore different control actions will be activated. But the result will be a crisp value which can be calculated by different approaches. The most common defuzzification methods are :

(i) Center of gravity (COG). This method computes the centroid of the area determined by the joint membership functions of fuzzy action,

(ii) Mean-of-maxima (MOM). This method calculates the arithmetic mean of all values with maximum membership, and

(iii) Height defuzzification (HD). This method computes the weighted sum of the height values of all membership functions associated with conclusion terms.

March et al.(1993) evaluated several defuzzification methods for industrial application based on memory requirement and speed of computation, as illustrated in Tables 3.1 and 3.2.

In this study, the method used is that of the most commonly used defuzzification , which is the

Table 3.1 Comparison between some defuzzification methods based on speed and memory requirement (from March et al. 1993).

Characteristic of the crisp output	Defuzzification Method
Memory requirements [1] Speed [5] Product: Some output fluctuation	COG with singleton output membership function
Memory requirements [5] Speed [1] Product: Smooth output	COG with non-singleton output membership function
Memory requirements [2] Speed [4] Product: Some optimistic output	Left maximum
Memory requirements [4] Speed [3] Product: Very optimistic output	Right maximum
Memory requirements [4] Speed [3] Product: Optimistic output	Average of maxima
Memory requirements [3] Speed [4] Product: Some output fluctuation	Midpoint of maxima
Memory requirements [5] Speed [1] Product: Some output fluctuation	Medium
Note : Memory requirement ranges from [1] = low to [5] = high Computational speed ranges from [1] = slow to [5] = fast	

Table 3.2 The formulation of five defuzzification methods reported by the International Electro-technical Commission (IEC) : Measurement and Control (1993)

COG	$$Z = \frac{\int_{Min}^{Max} x\mu(x)dx}{\int_{Min}^{Max} \mu(x)dx}$$
COGS	$$Z = \frac{\sum_{i=1}^{p} x_i \times \mu(x_i)}{\sum_{i=1}^{p} \mu(x_i)}$$
COA	$$Z = x` \int_{Min}^{x`} \mu(x)dx = \int_{x`}^{Max} \mu(x)dx$$
RM	$$Z = \sup(x`) \times \mu(x`) = \sup_{x \varepsilon [Min,Max]} \mu(x)$$
LM	$$Z = \mathrm{Inf}(x`) \times \mu(x`) = \sup_{x \varepsilon [Min,Max]} \mu(x)$$

where:
- Z: Result of defuzzification
- x: output variable
- p: number of singletons
- μ: membership function after accumulation
- i: index
- Min: lower limit for defuzzification
- Max: upper limit for defuzzification
- Sup: largest value
- Inf: smallest value

centroid defuzzifer called center of gravity (COG) with singleton output defined as:

$$y = \frac{\sum_{i_1 i_2 \ldots i_n \in m} V \mu_\circ R_{i_1 \, i_2,\ldots i_n} (y_{i_1 \, i_2,\ldots i_n}) y_{i_1 \, i_2,\ldots i_n}}{\sum_{i_1 i_2 \ldots i_n \in m} V \mu_\circ R_{i_1 \, i_2,\ldots i_n} (y_{i_1 \, i_2,\ldots i_n})} \tag{3.16}$$

and,

$$m = (i_1 \; i_2 \ldots . i_n | \; i_j = 1,2 \ldots, N_j \; ; \; j = 1,2,\ldots . n) \tag{3.17}$$

When the fuzzy system is a single-input single-output (SISO), the fuzzy rule base is given by:

$$R_i = IF \; x \; is \; \mu_i, \; THEN \; y \; is \; C_i \; , i = 1,2,\ldots\ldots, N \tag{3.18}$$

and Eq. 3.16 then becomes:

$$y = f(x) = \sum_{i=1}^{N} [\frac{\mu_i (x)}{\sum_{i=1}^{N} \mu_i (x)}] . \; y_i \tag{3.19}$$

The antecedent (the rule's premise) describes to what degree the rule applies, while the conclusion (the rule's consequent) assigns a membership function to each of one or more output variables. Most tools for working with fuzzy expert systems allow more than one conclusion per rule. The set of rules in a fuzzy expert system is known as the rule base system.

3.4 FLC Models

There are four possible forms for construction of a fuzzy logic control system based on input-output relations. A FLC can be in the form of a single-input-single-output (SISO), double-input-single-output (DISO), multi-input-single-output (MISO) system, or multi-input- multi-output (MIMO) system. In this study, SISO, DISO, and MISO models will be applied for estimation of a curve number (CN). The theory for each of these three models is discussed below.

3.4.1. SISO Model

On the basis of a verbal description, which is called a linguistic model as introduced by Zadeh (1973), an overall fuzzy relation R for a single-input-single-output (SISO) system is created by the formula below, where X_i and Z_i stand for fuzzy (linguistic) inputs and outputs, respectively. Gupta et al. (1986) introduced an open-loop SISO model such that the output has no effect upon the input control actions. A block diagram of the fuzzy system for an open loop SISO model is shown in Fig 3.4a. Such a system, when described by fuzzy relations is called an open-input fuzzy system. The compositional rule relation used by Zadah (1973) and Mamdani (1974) is:

$$R = \mathop{\mathcal{L}}_{i=1}^{N} (X_i \rightarrow Z_i) \tag{3.20}$$

where $\mathcal{L}$ stands for an operation which interprets the sentence connective "ALSO". It is assumed that, the verbal description of the process behaviour contains N relations:

$$\begin{aligned} &R1 : IF\ X\ is\ (X_1)\ THEN\ Z\ is\ (Z_1) \\ &\qquad ALSO \\ &\qquad \cdots\cdots \\ &RN : IF\ X\ is\ (X_N)\ THEN\ Z\ is\ (Z_N) \end{aligned} \tag{3.21}$$

In enhanced form,

$$\begin{aligned} R1\ (u\ ,w) &= \min(X1(u),\ Z1(w)) \\ &\cdots\cdots\cdots\cdots \\ RN\ (u\ ,w) &= \min(XN(u),\ ZN(w)) \end{aligned} \tag{3.22}$$

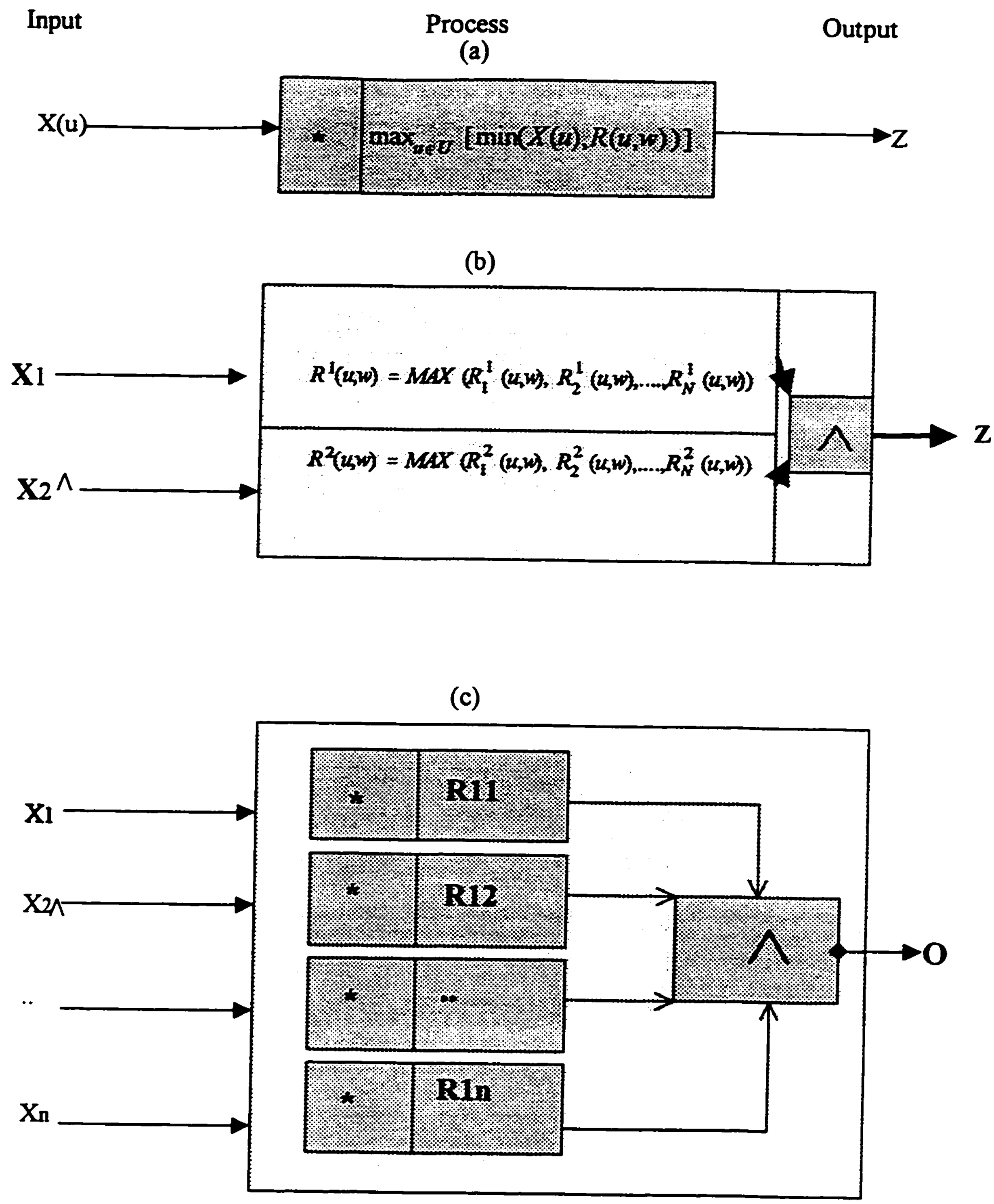


Fig 3.4 Fuzzy logic controller open loop model for (a):SISO; (b) DISO (c):MISO

where u and w stand for the universe of discourse of fuzzy inputs and outputs, respectively. The relation R should be obtained as a union of R1, R2, R3,....,RN since the sentence "also" is defined as the union:

$$R = R1 \cup R2 \cup, \ldots\ldots, \cup RN$$
$$R(u,w) = \max(R1(u,w), \ldots\ldots, RN(u,w)) \tag{3.23}$$

Min and max operators are chosen for intersection and union. The compositional rule of inference for approximate reasoning is suggested by Zadeh (1973). Max-min composition is chosen to infer a fuzzy conclusion Z to a fuzzy observation X. If the fuzzy relations R1, R2,....RN are created by application of some definition of fuzzy implication such as {[R1 = X1 x Z1],......,[RN = XN x ZN]} then:

$$Z = X \circ R \tag{3.24}$$

and,

$$Z(w) = \max_{u \in U} \left[\min(X(u), R(u,w))\right] \tag{3.25}$$

3.4.2 DISO Model

For the case of a double-input-single-output (DISO) controller, the method of infering the response to the inputs consists of extending the model which can be characterized by variables X^1 and X^2 as inputs and Z as output. A block diagram of the fuzzy system for an open loop DISO model is shown in Fig 3.4b. This model consists of creating the set of linguistic statements, called fuzzy rules, where fuzzy subsets of input and output variables are used as antecedents and consequents. If X^1_1, X^1_2 ,.......X^1_N is a fuzzy subset of X^1, and X^2_1, X^2_2 ,.......X^2_N are fuzzy subsets of X^2, and Z_1, Z_2,........Z_N are fuzzy subset of Z, then fuzzy relation R is defined by a set of fuzzy rules as follows:

$$
\begin{aligned}
&R1 : IF\ X^1\ IS\ X_1^1 \wedge X^2\ IS\ X_1^2\ THEN\ Z\ IS\ Z_1,\ ALSO\\
&R2 : IF\ X^1\ IS\ X_2^1 \wedge X^2\ IS\ X_2^2\ THEN\ Z\ IS\ Z_2,\ ALSO\\
&\quad .....\\
&Ri : IF\ X^1\ IS\ X_i^1 \wedge X^2\ IS\ X_j^2\ THEN\ Z\ IS\ Z_j,\ ALSO\\
&R1 : IF\ X^1\ IS\ X_N^1 \wedge X^2\ IS\ X_N^2\ THEN\ Z\ IS\ Z_N.
\end{aligned}
\tag{3.26}
$$

By normalizing the inputs and output in the same universe of discourse a method, to create the fuzzy relation R_i which represents the fuzzy implication, can be derived by the decomposition of the first rule into two parts. For the first variable (X^1):

$$R_i^1 = X_i^1 \circ Z_i \tag{3.27}$$

where $^\circ$ stands for the max-product operator and for the second variable,

$$R_i^2 = X_i^2 \times Z_i \tag{3.28}$$

Note that R^1_i and R^2_i represent partial relations, or sub-relations, and therefore the overall relation $R = R^1 \cup R^2$. The overall relation for the combination of the first variable X^1 and the output Z is therefore given by:

$$R^1 = R_1^1 \cup R_2^1 \cup, ..., \cup R_N^1 \tag{3.29}$$

and for the combination of the second variable X^2 and the output Z it is:

$$R^2 = R_1^2 \cup R_2^2 \cup, ..., \cup R_N^2 \tag{3.30}$$

By replacing the UNION operators with a fuzzy MAX operator the following is obtained:

$$R^1(u,w) = MAX\ (R_1^1\ (u,w),\ R_2^1\ (u,w),....,R_N^1\ (u,w))$$
$$R^2(u,w) = MAX\ (R_1^2\ (u,w),\ R_2^2\ (u,w),....,R_N^2\ (u,w)) \tag{3.31}$$

As a result, two overall rules coexist in this model and can be used to infer output Z by using the MIN-superposition of two relations with respect to the input X^1 and X^2. An alternative way to infer DISO is:

$$Z = MIN_{u\in U}(MAX\ MIN[X^1,\ R^1(u,w)],\ MAX\ MIN[X^2,\ R^2(u,w)]) \tag{3.32}$$

3.4.3 MISO Model

To describe the multi-input-single-output (MISO) system (Fig.3.4c), let the system's performance be given by N relations (R) and the system have M fuzzy inputs $X^{(1)}$, $X^{(2)}$, ..., $X^{(M)}$ and a single output O. Then Eq. 3.12 is rewritten as:

$$\begin{aligned} R_1 &= IF\ X^{(1)}\ is\ (X_1^{(1)}) \wedge X^{(2)}\ is\ (X_1^{(2)}).........\wedge X^{(M)}\ is\ (X_1^{(M)})\ THEN\ O\ is\ O1 \\ &\qquad ALSO \\ &\qquad \vdots \\ R_N &= IF\ X^{(1)}\ is\ (X_N^{(1)}) \wedge X^{(2)}\ is\ (X_N^{(2)}).........\wedge X^{(M)}\ is\ (X_N^{(M)})\ THEN\ O\ is\ ON \end{aligned} \tag{3.33}$$

where $X_1^{(1)}$, $X_1^{(2)}$,....,$X_1^{(M)}$, and O1 stand for the values of fuzzy inputs $X^{(1)}$, $X^{(2)}$,...,$X^{(M)}$ and fuzzy output O, respectively, when rule R_1 is created, and $X_N^{(1)}$, $X_N^{(2)}$,........, X_N (M) are input values, and ON is the output value used to create rule R_N. The variables $X^{(1)}$, $X^{(2)}$,...., $X^{(M)}$ are normalized to the same universe of discourse. The first and following rules are decomposed into M separate sub-rules and are given as:

$$R_1^{(1)} = X_1^{(1)} \times O1$$
$$\vdots$$
$$R_1^{(M)} = X_1^{(M)} \times O1$$
$$\vdots$$
$$R_N^{(1)} = X_N^{(1)} \times ON \tag{3.34}$$
$$\vdots$$
$$R_N^{(M)} = X_N^{(M)} \times ON$$

In enhanced form, Eq. 3.34 can be rewritten as:

$$R_1^{(1)}\ (u,w) = \min(X_1^{(1)}\ (u),\ O1(w))$$
$$\vdots$$
$$R_1^{(M)}\ (u,w) = \min(X_1^{(M)}(u),\ O1(w))$$
$$\vdots$$
$$R_N^{(1)}\ (u,w) = \min(X_N^{(1)}\ (u),\ O1(w)) \tag{3.35}$$
$$\vdots$$
$$R_N^{(M)}\ (u,w) = \min(X_N^{(M)}\ (u),\ O1(w))$$

where u and w stand for the universe of discourse for inputs, respectively. The M overall sub-rules are then given by:

$$R^{(1)} = R1(1) \cup R2(1) \cup \ldots\ldots \cup RN(1)$$
$$\ldots$$
$$R^{(M)} = R1(M) \cup R2(M) \cup \ldots\ldots \cup RN(M) \tag{3.36}$$

Eqs. 3.35 and 3.36 can be enhanced as follows:

$$\begin{aligned} R1^{(1)}(u,w) &= \max \langle R1^{(1)}(u,w),\ R2^{(1)}(u,w), \ldots RN^{(1)}(u,w) \rangle \\ &\ldots\ldots \\ R1^{(M)}(u,w) &= \max \langle R1^{(M)}(u,w),\ R2^{(M)}(u,w), \ldots RN^{(M)}(u,w) \rangle \end{aligned} \tag{3.37}$$

In this case, the fuzzy output of the MISO model is obtained using min-superposition of all the M outputs inferred from the sub-models:

$$O = \min \langle (X^{(1)} \circ R^{(1)}), \ldots, (X^{(M)} \circ R^{(M)}) \rangle \tag{3.38}$$

It should be noted that the projection applied in Eq. 3.38 which determines a one-dimensional fuzzy output will result in some loss of accuracy. In the extended form, Eq. 3.38 is rewritten as:

$$\begin{aligned} O(w) = \min \langle &[\max_u (\min (X^{(1)}(u),\ R^{(1)}(u,\ w)))] .. \\ &, \ldots, [\max_u (\min (X^{(M)}(u),\ R^{(M)}(u,\ w)))] \rangle \end{aligned} \tag{3.39}$$

3.5 Fuzzy Controller Requirements

In fuzzy logic inference, and particularly in fuzzy-logic control, one may run into difficulties unless certain conditions are satisfied in the associated knowledge base.

FLC requires a 5-step sequence of principles and procedures for designing a fuzzy logic rule-based system as:

(i) Analyze and partition the control system:
  - (a) Identify inputs and outputs,
  - (b) Analyze and simplify the problem, and
  - (c) Identify fuzzy units (i.e. linguistically oriented adjectives).

(ii) Define input and output surfaces:
  - (a) Specify universe of discourse,
  - (b) Scale universe of discourse,

(c) Determine number and distribution of membership functions, and

(d) Compare the deffuzification methods.

(iii) Write rules:

(a) Write obvious rules,

(b) Write less obvious rules,

(c) Consider special cases, and

(d) Rule tuning (alpha-cut, contribution weights, compensatory operators).

(iv) Observe model behavior, begin verification and tune as needed:

(a) Examine output results, and

(b) Examine control surface.

(v) Optimize system for target platform:

(a) Implementation schemes.

3.5.1 Analyze and Partition the Control System

As in many designs, a starting point is to first describe the overall system (both the fuzzy and the non-fuzzy variables) to be controlled. Specifically, it is necessary to specify what inputs go to the system, and what outputs come out However, since design is an iterative process, it may be found necessary and/or beneficial to go back and forth between steps to simplify the problem.

3.5.2 Define Input and Output Surfaces

Once the inputs and output have been identified, the universe of discourse and its scale have to be considered. The universe of discourse has to be an optimum for an input. If, the universe of discourse is too large, the response will be large, resulting in a flat response.

An output membership function manifests itself in the defuzzification process. A large output membership function will overpower others by lending its entire mass to the defuzzification process.

Determination of the number and distribution of membership functions depends on the sensitivity of the output to the input. Too many membership functions can rapidly fire different rules for small changes in input values, resulting in large output changes which in turn can cause system instability. In practical application, the number of membership functions ranges from 3 to 9 with appropriate overlap of memberships .

3.5.2.1 Overlap Indices

March et al. (1993) has proposed two indices to describe the overlap of membership functions quantitatively as:

$$Overlap\ Ratio = \frac{Overlap\ scope}{Adjacent\ scope} \tag{3.40a}$$

and,

$$\begin{aligned} Overlap\ Robustness &= \frac{Area\ of\ summed\ overlap}{Maximum\ area\ of\ summed\ overlap} \\ &= \frac{\int_U^L (\mu_{A_1} + \mu_{A_2})dx}{2[U-L]} \end{aligned} \tag{3.40b}$$

A guideline for an overlap ratio indicates that this should be kept between 0.2 and 0.6. The value of overlap robustness is usually greater than that of overlap ratio: it should be kept between 0.3 and 0.7. For smooth operation of a fuzzy control model the desirable overlap ratio is about 0.33 and overlap robustness about 0.5.

3.5.3 Rule-Base System

From a knowledge acquisition point of view, a fuzzy expert system is in the form of a rule-based system. The rule-base system contains expert knowledge in the form of linguistic expressions with some linguistic modifiers. Of interest in the use of linguistic variables in the management of uncertainty, is the capacity for changing the intensity of the characteristic of the variable by means of linguistic modifiers (Zadeh 1973). Zadeh (1996) stated that *fuzzy logic = computing with linguistic words*. A linguistic modifier is an operator M which provides a new characterization M(A) of a fuzzy set A in the form of a mathematical function F in which:

$$\forall u \in U \ \mu_{M(A)}(u) = F(\mu_A(u)) \tag{3.41}$$

There are several linguistic modifiers such as "very", "strongly", "really, "moderately", "not very", "relatively", etc. However, only four simple modifiers $\mu_M(x)$ =[very, low, medium, high] will be used.

Three kinds of production rules have been defined:

(i) Simple rule. This kind of rule consists of a SISO model in the form of:

IF <antecedent> THEN <consequent> <CF>.

(i) Multiple proposition rule. This is composed of several simple rules and can be either a MISO or a MIMO in the form of:

IF < operator> <antecedent 1>

<antecedent 2>

....

<antecedent n>

THEN <consequent/s> <CF>

The operators may be one of, or a combination of such operators as AND, OR, ALSO, or AVG.

(iii) Gradual rules. Sometimes two data sets are closely connected. The relation between such gradual knowledge can be formulated with a gradual rule. In such a rule, a little modification in the antecedent part entails some modification in the consequent part. By using classical rules, the expert has to introduce in the rule-base all of the possible linguistic variable's combinations to formalize the relation between the linguistic variables of the antecedent and the consequent part. By using gradual rules, it is only necessary to use judicious linguistic variables and values with two key words "more" and/or "less". A gradual rule then is expressed by:

IF <more/less><antecedent> THEN <more/less><consequent> <CF>

For each rule there is a certainty factor <FC> characterized by key words which represent the importance of degree in the rules such as the following associations. The factors of certainty may be presented by <FC> as:

Never	0	Very uncommon	0.1
Uncommon	0.2	Not usual	0.3
Sometimes	0.4	Neutral	0.5
Quite common	0.6	Common	0.7
Very common	0.8	Principally	0.9
		Always	1.0

The factors of certainty are illustrated in Fig 3.5.

3.5.3.1 Writing Rules

To write rules, one must first encode the knowledge describing the system behavior. This encoding results from observations of:

(i) System operators,

(ii) Interviewing experts, and/or

(iii) Using existing control surfaces to describe the model. This generally provides a good description of the model.

FLC behavior is defined by rules which map input labels to associated output labels by a knowledge base system.

A rule-base should meet the following conditions:

(i) The rule base should be "complete",

(ii) The rules should not "interact",

(iii) The rules should be "consistent",

(iv) The inferences should be "continuous", and

(v) The rule-based system should be "robust" and "stable".

The first phase of rule encoding involves describing all the rules that are practical to describe a system. It should entail writing rules based on the visibility of rules. On the other hand, there are rules that are obvious, yet intuitively correct. The second phase of writing rules is after the model has been simulated and its behavior observed. Individual rules are then tuned

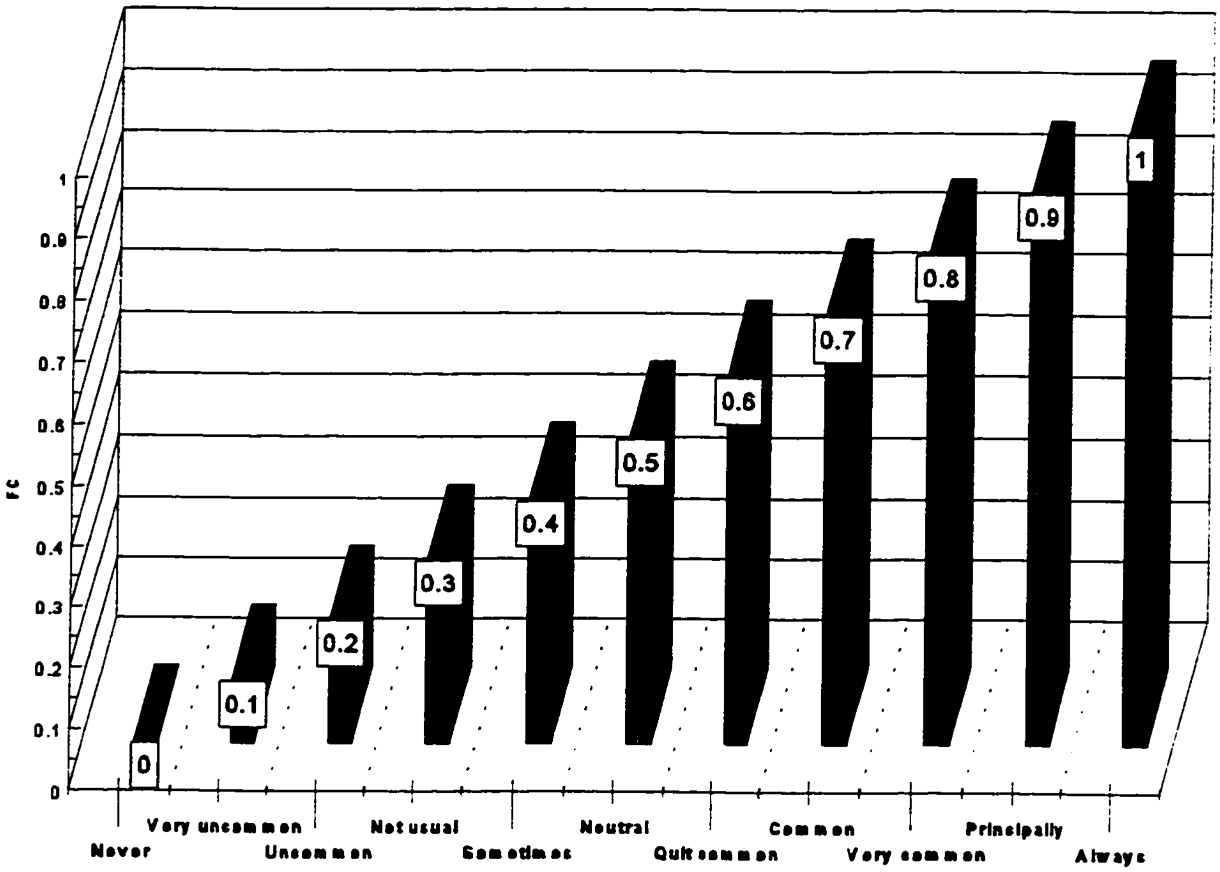


Fig 3.5 The relationship between factor of certainty (FC) and its linguistic equivalent.

so that their contribution to the control surface reflects more accurately the desired behavior of the system being modeled. There are three areas to consider when tuning: compensatory operators, alpha cuts, and contribution weights (techniques which depend on the desired model accuracy).

3.5.4 Observed Model Behavior and Verification

The model characteristics that need to be verified are:

(i) Output values: the defuzzified values returned by the model should be checked for accuracy.

(ii) Control surface: this consists of mapping of inputs to output as in a control surface and is possible only for a DISO (two inputs and one output system) models for which the control surface can be visualized in 2- or 3-dimensions, respectively.

(iii) Time simulation: the behavior of the model over time must be observed for sensitivity and stability.

3.5.5 Rule Tuning

Once the model is running and the simulated output is determined, the output behavior can be modified by tuning the individual rules to reflect more accurately the desired system output. Techniques that can be used for tuning are alpha cuts, contribution weights, and compensatory operators which are described below.

3.5.5.1 Alpha Cuts

An alpha cut is a process in which a designer specifies a threshold under which specified fuzzy properties are evaluated; i.e., which rules are permitted to contribute to an inference process. Two types of alpha cuts are:

(i) Fuzzy Set-Level. This type of alpha-cut is applied at a level of individual fuzzy sets and serves globally for the whole system.

(ii) Rule-Level. This type is applied at the rule level. If the truth value of the premise of a rule falls below this type of alpha cut, the rule strength is set to zero. This means that the rule will not fire in control action.

3.5.5.2 Contribution Weights

Not all fuzzy rules are created equal during operation. In traditional fuzzy logic, rules are selected and fired based on compatibility between the rules and the current input data vector. The strength of a rule's contribution to a consequent fuzzy region depends solely on this level of compatibility. To provide the engineer with means to prioritize rules, a multiplicative weighting factor can be assigned to each rule. These weights are called contribution weights since they adjust the contribution of each of the rules to the final model solution. This weight normally lies in the interval [0, 1].

3.5.5.3 Assertion

Assertions are consequents without antecedents. An assertion in a fuzzy model can be conditional or unconditional. An unconditional assertion establishes a fuzzy region that acts as the limiting boundary for the consequent solution space. For example, the statements "saturation situation has maximum curve number" or " very high sandy soil has minimum curve number"is an unconditional assertion. When an unconditional assertion using fuzzy set W is applied to fuzzy region S, the consequent region S*, is constrained as follows:

$$S^{*} = \min[\mu_W(x),\ \mu_S(x)] \tag{3.42}$$

The result of defuzzification for an unconditional assertion when the assertion is *minima*l should be less then the consequent that is obtained by an ordinary COG method, or if the assertion is *maximal* the consequent should be greater than the COG defuzzification result.

3.5.5.4 Compensatory Operators

Connective operators compensate for Zadeh's strict rules of combination for fuzzy intersection [AND] and union [OR]. The type of compensatory operators used in the rule evaluation process impacts the way in which the control surface is created.

One possible problem with the min/max operators is that if rules with many antecedents are written, the AND operator moves the truth value to zero and OR moves the truth to unity. If this type of behavior is not acceptable within a system, an average or sum operator may be more appropriately used.

3.6 Knowledge Translation

In the development of a fuzzy logic application there are several different languages and shell expert tools that are used in the field of fuzzy reasoning strategy. The choice of the development environment will determine the structure that will go into the formalization.

The most prevalent of the languages that are used in the development of fuzzy logic control are described below:

(i) Procedural and logical languages. The most prevalent procedural language is the family of C languages. C languages are midway between algebraic languages, such as FORTRAN, BASIC, ADA, etc., and logical languages such as LISP (McCarthy et al. 1965) and PROLOG. Algebraic languages are known as procedural languages, because of the methodology of solving the problem by algorithm. On the other hand, logical languages deal with problems with strategies such as backward or forward reasoning.

The knowledge representation of a fuzzy process is composed of data, rules and procedures (Ross 1992). C is a scientific programing language, and in some case is a lower level language, than a logical language. More recently, C was integrated to produce object oriented programming capability using C++ libraries (Rao and Rao 1993).

Fuzzy logic acts as an algorithm for solving a problem. Fuzzy logic therefore has a procedural structure from a programing point of view. In addition, fuzzy logic uses a forward reasoning strategy as in an expert system. C language has the capacity to deal with both characteristics of procedural and logical programming.

(ii) Fuzzy expert shells: Expert shells may be classified according to the method of knowledge output. In general, there are different classes of fuzzy expert shell knowledge generator :

(a) Code generators. As an example FUZZLE is a fuzzy system development shell for PCs that helps the user understand the advantages and limitations of this new technology via its straightforward and simple work flow. The software is fully supported with graphics displays and mouse functions. One of the outputs is a source code in C or FORTRAN language that can be converted into an executable code and attached to an application environment. In addition, FUZZLE has its own execution module with graphics support which does not require any programming, compilation or linking. Using this option, the user can obtain inference results directly from the FUZZLE shell by entering data via the keyboard or by use of external data files. Once the inference engine is validated, the executable portion of the software can be extracted by a click of a button and it can be customized for special purpose applications. The final product, which is stripped from FUZZLE related functions and screen images, is ready for distribution and it is royalty-free. Thus, it is possible to not only develop a fuzzy inference engine, but also to build software using FUZZLE. This is all possible without knowing any programming language or compiling any program segments.

(b) Knowledge-based generator. This kind of expert shell generates only the knowledge-based (rule-based) code rather than exact code. The amount of information depends on the design purpose. For instance, Fuzzy Knowledge Builder (FKB) developed by McNiell (1994) for a PC platform is a software product of use to any fuzzy system designer. It is a graphics-based rule and fuzzy set editor. It allows simple definitions of the verbal meanings of fuzzy sets. Rules are displayed and edited as a graphics multi-dimensional matrix. Fuzzy sets are displayed and edited as graphic line drawings. Output after editing is a file to be included in the application source code. This file contains the rules and fuzzy sets captured from the expert and this is needed for the construction of the fuzzy associative memory or fuzzy estimation surface in any application. The program provides an intuitively helpful interface for capturing the expert judgements needed to build a fuzzy system.

Fuzzy systems use fuzzy estimation and control transforms. Input and output fuzzy sets and rules for mapping from input to output make up the knowledge base. But, the main problem for this kind of expert shell is the generation of a rule-based system based on random rule generation. For example, for a very sensitive output as in this study, the generated rule base is full of uncertainties. For example, for a FLC with five inputs with this amount of fuzzy membership functions the number of possible rules which can be generated is equal 5*5*5*8*3 = 3000. This knowledge base is very large to manipulate with a PC computer. In other words, the use of this kind of shell is limited in a small computer environment.

(iii) Combined fuzzy language and expert shell. As an example of this mixed language, the expert shell Fuzzy Programming Language (FPL) was developed by Togai InfraLogic (1993) specifically for the implementation of fuzzy logic systems. Consisting of fifteen different objects, FPL provides ease and flexibility in defining fuzzy systems. TILShell accelerates the fuzzy system design process by allowing the user to define and manipulate the objects of FPL graphically rather than as text. The debugging and tuning tools of TILShell interact directly with the FPL file which allows the fuzzy system to be thoroughly tested before turning it into an executable form, thus saving considerable time and effort in the development process. The Fuzzy-C Compiler converts a fuzzy system from FPL into C source code which then may be compiled and linked with other modules to produce an executable application. The Fuzzy-C Compiler is specifically designed to provide the maximum possible flexibility in terms of trading between speed, space, and precision of the output C source code. The most important interaction between the fuzzy system and the rest of the program, from the point of view of the user, are the calls to the fuzzy system by the rest of the program. FPL makes this interface very simple. The program simply calls the name of the Project object specified in the FPL file with the appropriate parameters as if it were a C routine (which by this time it has been compiled to).

4. CURVE NUMBER COMPUTATION

Fuzzy logic controllers mimic human knowledge by describing control strategies using linguistic rules. Clearly, the control rules are model-free: no matter how mathematically difficult the process is, an experienced operator can still generate some control rules. Fuzzy logic controllers are most applicable to non-linear, dynamic, and ill-understood processes.

In particular, fuzzy rules may have different semantics, which lead to different choices concerning the multiple-valued connective used to model the rules.

In this study, based on available information and the theoretical relationships between parameters, it is assumed that the rule-based system is purely gradual with uncertain conclusion parts. Purely gradual rules are of the form "the more X is A, the more Y is B" which qualitatively describes a relation between the values of X and Y without certainty.

In this study it is assumed that the curve number (CN) is a function of soil texture (S), cover density (C), moisture content (M), and land use (L). This chapter details the application of FLC models for all of these variables to determine the curve number (CN).

The FLC design in this study will be implemented in three different stages in terms of the FLC models for which the theory was presented in Ch.3. The first model is a SISO and is a preliminary model for % clay and % sand as individual inputs with curve number as the output. The second FLC model is a DISO applied to soil texture versus each of the parameters in turn to improve the relationship between an individual parameter and the curve number. The final, complete model is in the form of a multi-input-single-output (MISO).

4.1 SISO Model

With reference to the concepts of fuzzy set theory (described in detail in Ch.3), this section considers the nature of fuzzy data. As a start, a distinction is made between the way in which we quantify between crisp data and fuzzy data. For example, in standard science and technology for time when specified as a specific moment the value is a crisp (and definite) number (e.g. one second, two hours, one thousand years, etc.). However, if time is specified in terms such as "very long time", "long time", "short time", "very short time", etc. these adjectives are not crisp (i.e., indefinite) and are described as being fuzzy. By the same argument, physical and chemical soil properties such as infiltration, hydraulic conductivity, soil texture, permeability, soil moisture, etc. can be classified as fuzzy (and specified as fuzzy data ranging in value). McBratney and De Gruijiter (1992) and Burrough et al. (1992) investigated the use of fuzzy set theory for soil classification. The purpose of classification was to reduce a complex system, represented by some sets of data, into explicitly defined classes. Observations are grouped into continuous classes (fuzzy) rather than exactly defined (hard) classes.

In this section, it is assumed that the relationship between soil texture and curve number is not well known. However, the CN index can be specified for specific unique locations in a watershed. This process demonstrates considerable non-linearity with respect to different regions of soil classes within a soil classification diagram.

Let us represent the effects of soil texture (% sand and %clay) on curve number as {CNij}, where *i* denotes the i-th step of % sand and j denotes the j-th step of %clay. Also let the set of membership functions for % sand, % clay, and curve number be denoted by Si, Cj, and CN, respectively. For i = j = 10,

$$\begin{aligned} S &= [S10\ S20\ S30\ S40\ S50\ S60\ S70\ S80\ S90\ S100] \\ C &= [C10\ C20\ C30\ C40\ C50\ C60\ C70\ C80\ C90\ C100] \\ CN &= [CN10\ CN20\ CN30\ CN40\ CN50\ CN60\ CN70\ CN80\ CN90\ CN100] \end{aligned} \tag{4.1}$$

Based on the logic of the problem % sand has the effect of a decreasing distribution function (DDF) while % clay has the effect of an increasing distribution function (IDF), with both defined on μ : [0, 1]. At this time, we are using the following membership functions :

$$\mu_{Clay}(x) = C_i^{(1/2)} \quad , \quad \mu_{Sand}(x) = [1 - S_i^{(1/2)}] \tag{4.2}$$

where C_i and S_i represent % clay and % sand, respectively each with a membership function μ (x). Depending on the process characteristics, the process regions can be categorized as very low CN [CN10, CN20], low CN [CN30,CN40], medium CN [CN50,CN60], high CN [CN70, CN80], and very high CN [CN90, CN100].

By applying a SISO FLC model with inputs of soil texture (% sand and % clay individually) with a very small rule-base that contains only five rules (in the form of Eq. 3-21) then:

R1: IF the Clay is Very Low [C10, C20] THEN the CN is Very Low ALSO
R2: IF the Clay is Low [C30, C40] THEN the CN is Low ALSO
R3: IF the Clay is Medium [C50, C60] THEN the CN is Medium ALSO *(4.3)*
R4: IF the Clay is Low [C70, C80] THEN the CN is High ALSO
R5: IF the Clay is Very High [C90, C100] THEN the CN is Very High ALSO

and for % sand and curve number similar rules can be constructed:

R1: IF the Sand is Very Low [S10, S20] THEN the CN is Very High ALSO
R2: IF the Sand is Low [S30, S40] THEN the CN is High ALSO
R3: IF the Sand is Medium [S50, S60] THEN the CN is Medium ALSO *(4.4)*
R4: IF the Sand is Low [S70, S80] THEN the CN is Low ALSO
R5: IF the Sand is Very High [S90, S100] THEN the CN is Very Low ALSO

By following the methodology of coding the SISO as an open-FLC (as presented in Ch.3) the ALSO is a union operator (as in Eq. 3.3) which translates the fuzzy membership function to a crisp value by application of Eqs. 3.22, 3.23, 3.24, and 3.25. The input and output of such a model is given in Fig 4.1. This figure represents the relationship between each of % sand and % clay versus curve number based on the assumptions given above. Fig 4.1 shows that the effect of % clay on the curve number appears as a complement of the effect of % sand. In other words, clay has an inverse effect on the curve number such that:

$$CN_{Clay} = 100 - CN_{Sand} \tag{4.5}$$

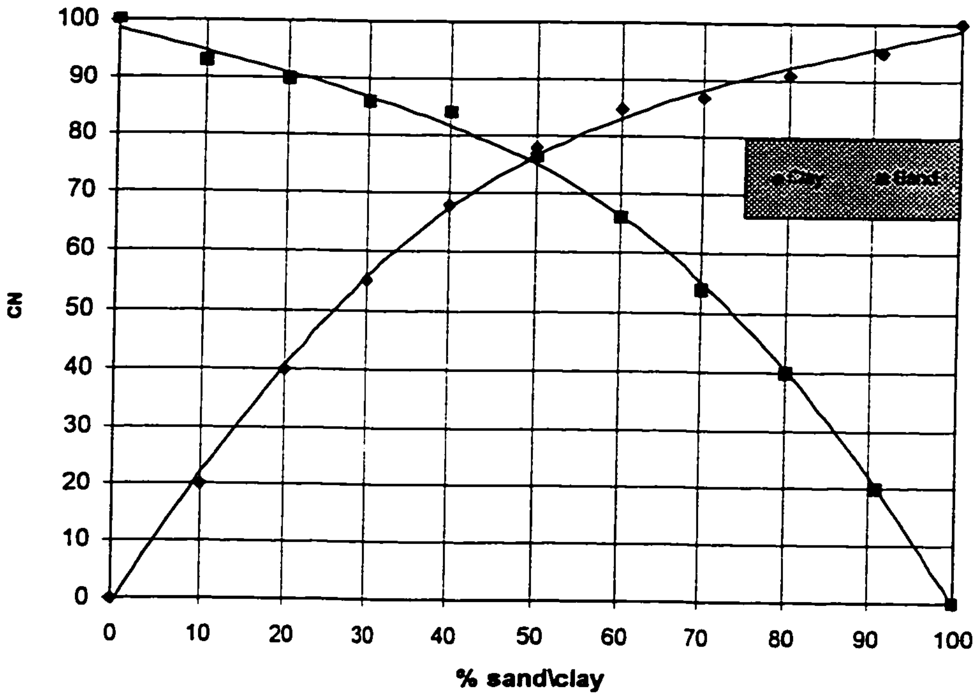


Fig 4.1 Relationship between % sand, % clay and curve number (CN)

4.2 DISO Models

Double-input-single-output (DISO) FLCs were constructed for % sand versus each of the other parameters with the three outputs used as input to a multidimensional-input-single-output MISO for final estimation of CN. In a DISO, one way to recognize the possible rule construction and visualization is by applying an n-dimensional matrix, called a fuzzy associative memory (FAM), for each two variables. A FAM is an n-dimensional table where each dimension corresponds to one input universe of rules. The *i*th dimension of the table is indexed by the fuzzy sets that comprise the decomposition of the *i*th input domain.

For example, by using the FAM matrix, 25 possible combination rules relating % sand and % cover density to CN is derived in rules system cells. Each cell in the FAM represents one possible rule. The result of the defuzzification is a crisp number which is presented in the form of a control surface in three-dimensional form. All of the stages involved in the FLC model described in Ch.3 are considered in model development. Only the first DISO FLC (sand and clay inputs) model is presented in detail as representative of the other component DISO models.

4.2.1 Soil Texture

Soil properties markedly influence the process of generation of runoff from rainfall and must be considered directly in methods of runoff estimation. The relevant soil properties are commonly represented by a hydrologic parameter which indicates the runoff potential of a soil and is the qualitative basis for the classification of all soils into four groups (A, B, C or D). The classification is broad but the groups can be divided into subgroups if desired. Wood and Blackburn (1984) evaluated these four hydrologic soil groups and reported that the hydrologic soil group classification system provides a poor basis for estimating infiltration rates on rangelands but that modifying them may even accentuate the prediction error. Schmidt and Schulze (1987b) adapted the SCS method by changing the four basic soil hydrology groups for Southern African soils. Three intermediate soil groups have been used in the classification of soil forms and series. Additional groups are A/B, B/C and C/D, giving a total of seven soil groups (A, A/B, B, B/C, C, C/D, D).

In this study fuzzy logic facilitates change from division into hydrologic soil groups to a continuum classification based on the two parameters, % sand and % clay.

It is assumed that the effects of % sand and % clay on the estimation of CN are the same but the relations are inverse (CN = $\int$(%clay) and CN = $\int$(1/ %sand) as formulated in Eq .4.5 as a result of the previous SISO model. Also it is assumed that the soil texture is continuous by use of five fuzzy labels. Soil can range from very sandy (sand > 80%) to heavy clay (sand <30%) based on a rectangular or triangular soil classification.

In considering soil texture alone, an assumption is made of no soil cover and an initially dry soil. In this case CN is a function of soil texture only. Soil texture is composed of three elements viz. sand, silt, and clay. These basic elements are classified in a rectangular soil classification that presents twelve soil classes. Infiltration of water into the soil and the transmission rate for these classes range from VERY RAPID for sandy soil to VERY SLOW for clay soil. The curve number is based on the infiltration capacity and runoff potential of each of the soils. The right-angle soil classification was used as a fuzzy associated matrix (FAM) for sand, clay, and the curve number. The hypothetical representation of the fuzzy membership function and, two-dimensional matrix for the rectangular soil classification is illustrated in Fig. 4.2 As can be seen, the extremes of the input universes of discourse corresponding to sand and clay are [0 - 100]. Based on the definition of curve number the universe of discourse of the output value is [0 - 100]. All domains under consideration were normalized for purposes of programming.

4.2.1.1 Terms and Membership Function

The choice of the number of terms of the linguistic variables as well as the shape and position of the corresponding membership function is essential for the next design step. The magnitude of the task of formulation of the rule-base depends exponentially on the number of terms or adjectives. A larger number of terms enables smoother control actions, but leads to a significant increase in the number of rules. In expert systems this is called combinatoric explosion.

An example of combinatoric explosion is the following. If the input variable e(k) distinguishes between three terms, and two previously obtained system deviations e(k-1) and e(k-2) are also used, then the number of possible rules equals $3 \times 3 \times 3 = 27$ for a complete

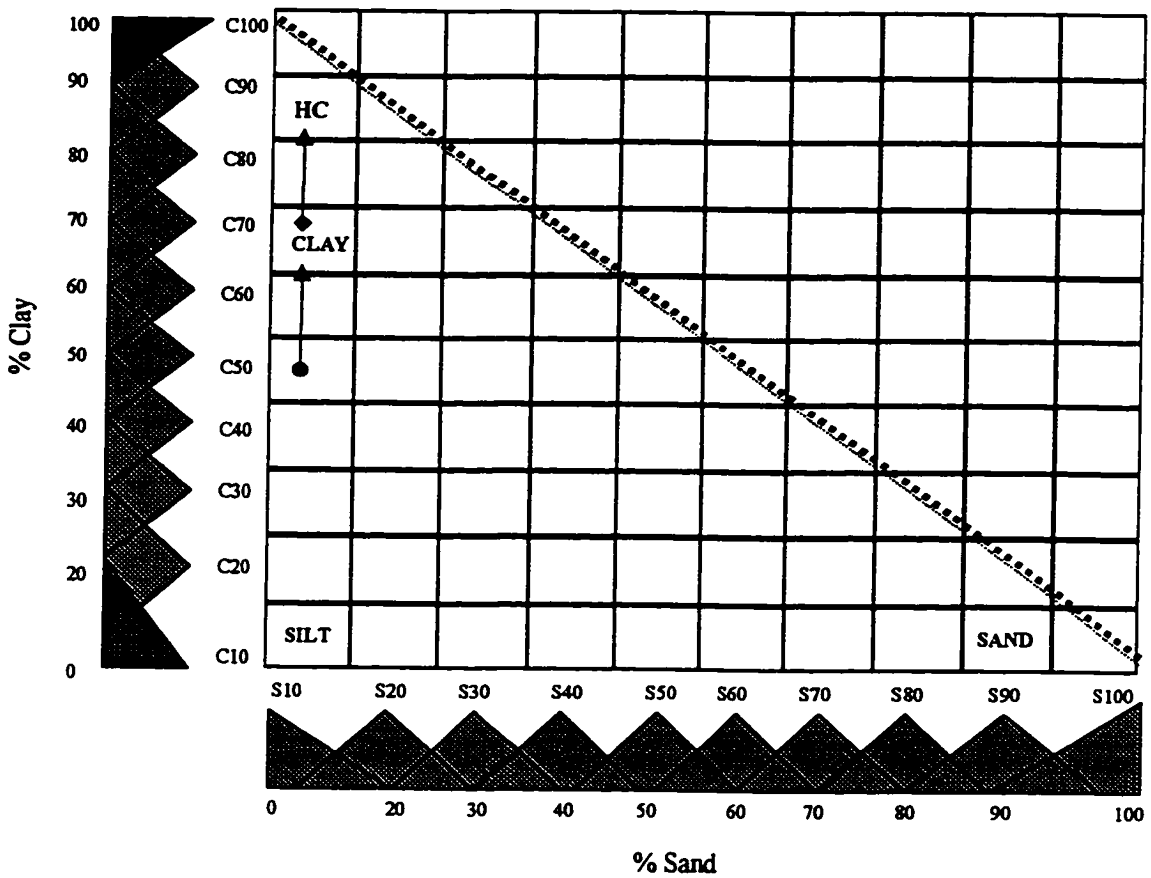


Fig 4.2 Fuzzification of cell groups of % sand, % clay, and curve number based on rectangular soil classification.

and consistent rule-base. If the number of distinguished terms is increased to five, 5 × 5 × 5 = 125 rules are necessary.

Therefore a sensible compromise between accuracy and computing effort has to be determined for each application. During an examination of the form of the membership function, the normal distribution form Eq. 3.7 and the triangular form Eq. 3.9 were selected. The 3 × 3 FAM for sand, clay, and CN was applied for both membership function shapes, assuming the fuzzy interval to be the same for both. The results are presented in Figs 4.3 and 4.4 for the triangular and the normal distribution, respectively. As may be noted there is no considerable visual difference between Figs 4.3 and 4.4. In other words, these attributes are not sensitive to the membership function shape.

However, based on the previous SISO test, the three adjectives fuzzy membership function created an inadequate result as shown in Figs 4.3. and 4.4. It was therefore decided to increase the number of adjectives from three to ten to reduce the interval of simultaneous partial truth of several rules (normally four rules). The triangular decomposition membership function of inputs and output values are presented in Tables 4.1, 4.2, and 4.3, respectively.

However, some basic rules concerning shape, size and position also must be considered. A first distribution is usually equally distributed. The border terms of the control differences have to include the ranges of disturbance or larger system deviations in the whole measuring range. If the center of gravity is calculated for defuzzification in control applications, then the border terms of the manipulated variables have to be chosen so that the center and not the extreme equals the limit of the manipulated variable.

4.2.1.2 Rule-Base

For generation and checking of rule-bases concerning completeness and consistency, a rule matrix is an adequate representation scheme, especially for mappings which easily can be visualized in a three-dimensional diagram. But this is in contrast to the wish of realizing complex operator strategies which would lead to a higher controller input dimension.

If the rule-base is very sensitive a higher input dimension, taking more previous values into account is an adequate choice. In most cases two sampling steps are sufficient, taking the curvature of the input shape implicitly into account. However, as mentioned earlier, this

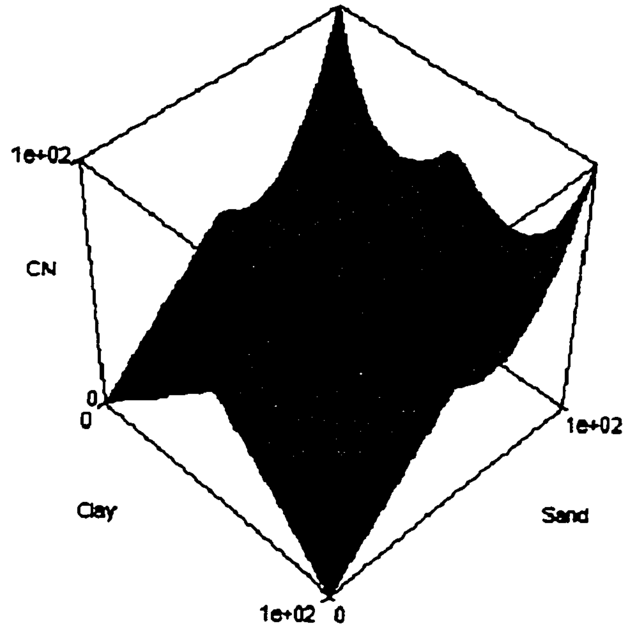


Fig 4.3. Control surface for CN based on a triangular membership function

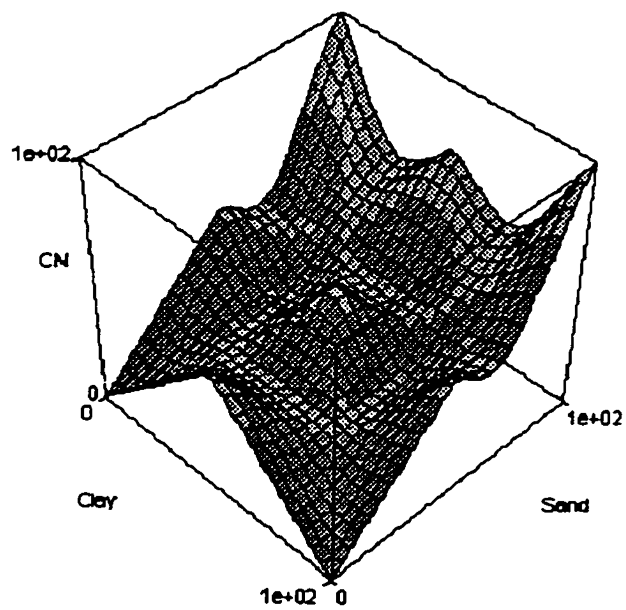


Fig 4.4. Control surface for CN based on a normal distribution membership function

Table 4.1 Adjectives for curve number and related membership function.

Adjectives	a_1 Crisp	a_1 $\mu(x)$	a_2 Crisp	a_2 $\mu(x)$	a_3 Crisp	a_3 $\mu(x)$
CN10	-		6.3	1.0	11.7	0.0
CN20	3.5	0.0	13.0	1.0	23.0	0.0
CN30	11.0	0.0	24.0	1.0	33.3	0.0
CN40	22.4	0.0	34.5	1.0	44.5	0.0
CN50	33.3	0.0	45.3	1.0	55.7	0.0
CN60	44.3	0.0	56.5	1.0	66.7	0.0
CN70	55.7	0.0	67.2	1.0	77.6	0.0
CN80	66.7	0.0	78.6	1.0	91.0	0.0
CN90	77.6	0.0	89.8	1.0	100	0.0
CN100	91.3	0.0	100	1.0	-	

Table 4.2 Adjectives for % sand and related membership function .

Adjectives	a_1 Crisp	a_1 $\mu(x)$	a_2 Crisp	a_2 $\mu(x)$	a_3 Crisp	a_3 $\mu(x)$
S10	-		4.5	1.0	11.0	0.0
S20	2.3	0.0	11.0	1.0	22.4	0.0
S30	11.0	0.0	22.4	1.0	33.3	0.0
S40	22.6	0.0	33.3	1.0	44.3	0.0
S50	35.3	0.0	45.3	1.0	57.7	0.0
S60	45.3	0.0	55.7	1.0	67.7	0.0
S70	55.7	0.0	68.7	1.0	77.6	0.0
S80	66.7	0.0	77.6	1.0	90.8	0.0
S90	77.9	0.0	90.0	1.0	100	0.0
S100	89.8	0.0	100	1.0	-	

Table 4.3 Adjectives for % clay and related membership function.

Adjectives	a1 Crisp	a1 $\mu(x)$	a2 Crisp	a2 $\mu(x)$	a3 Crisp	a3 $\mu(x)$
C10	-		0.0	1.0	15.0	0.0
C20	0.0	0.0	15.0	1.0	22.4	0.0
C30	15.0	0.0	22.4	1.0	33.3	0.0
C40	22.4	0.0	33.3	1.0	44.3	0.0
C50	33.3	0.0	44.3	1.0	55.7	0.0
C60	44.3	0.0	55.7	1.0	66.4	0.0
C70	55.7	0.0	66.7	1.0	77.6	0.0
C80	66.7	0.0	77.6	1.0	89.8	0.0
C90	77.9	0.0	89.0	1.0	100	0.0
C100	89.3	0.0	100	1.0	-	

increases the number of possible rules very quickly.

The relation between the inputs and output cannot be derived from a theoretical study using previous knowledge. However, there is a logical relationship among inputs and output. Where terms like "sandy soil", "heavy clay"...etc. are used to describe the fuzzy variables, the control reaction will be the curve number (CN). As mentioned previously, because of the partial matching attribute of fuzzy control rules, and the fact that the preconditions of rules overlap, up to four rules can fire at the same time. The fuzzification of the cell groups of the 10 x 10 FAM matrix and the triangular membership function are presented in Fig 4.5. This produces 100 rules in the form of the "*IF...AND...THEN*" rule-based system strategy, as presented in Table 4.4.

Assuming that in the inference sub-process, the Min-Max operator is used as the rule evaluator (as described in Ch.3), the methodology used in deciding what control action should be taken results in the firing of four rules. This combination of rules produces a non-fuzzy action or crisp value of output which is calculated using the COG defuzzification method (Eq. 3.16).

The three-dimensional control surface of the FAM is presented in Fig 4.6. This figure is a product of COG defuzzification and presents the interaction of sand versus clay to produce the curve number control surface. All possible rules are used in a wide range of test data across the universe of discourse. Fig 4.6 presents the control surface which results from the mapping of inputs (% sand and % clay) to output (curve number). As expected, the estimated maximum value of the curve number occurs with heavy clay and the minimum value with soil high in sand content. As may be seen the control surface shows the gradual effect of changes in the inputs on the resulting output. It is noted that the rules used to generate this surface (Table 4.4) are not based on experimential data, as in control applications of fuzzy logic control. These rules are based on logical relationships between the inputs and the output space which are applied in a SISO model (Sec. 4.1) as a purely gradual process with an uncertain conclusion part. This type of control surface does not need a compatibility index (CI) verification, as the compatibility index (CI) of inputs and the output space can be readily verified visually .

C100	CN100									
C90										
C80										
C70										
C60										
C50										
C40										
C30										
C20										
C10	CN90									CN10
	S10	S20	S30	S40	S50	S60	S70	S80	S90	S100

Fig 4.5 FAM matrix (10 x 10) for Sand (S), Clay (C), and curve number (CN)

Table 4.4 Rules-based system for sand, clay, and curve number (CN) based on 10 x 10 FAM matrix.

IF Sand IS S10 AND Clay IS C10 THEN CN IS CN90	IF Sand IS S10 AND Clay IS C60 THEN CN IS CN80
IF Sand IS S20 AND Clay IS C10 THEN CN IS CN90	IF Sand IS S20 AND Clay IS C60 THEN CN IS CN80
IF Sand IS S30 AND Clay IS C10 THEN CN IS CN80	IF Sand IS S30 AND Clay IS C60 THEN CN IS CN70
IF Sand IS S40 AND Clay IS C10 THEN CN IS CN70	IF Sand IS S40 AND Clay IS C60 THEN CN IS CN70
IF Sand IS S50 AND Clay IS C10 THEN CN IS CN70	IF Sand IS S50 AND Clay IS C60 THEN CN IS CN80
IF Sand IS S60 AND Clay IS C10 THEN CN IS CN60	IF Sand IS S60 AND Clay IS C60 THEN CN IS CN60
IF Sand IS S70 AND Clay IS C10 THEN CN IS CN40	IF Sand IS S70 AND Clay IS C60 THEN CN IS CN50
IF Sand IS S80 AND Clay IS C10 THEN CN IS CN40	IF Sand IS S80 AND Clay IS C60 THEN CN IS CN40
IF Sand IS S90 AND Clay IS C10 THEN CN IS CN10	IF Sand IS S90 AND Clay IS C60 THEN CN IS CN10
IF Sand IS S10 AND Clay IS C10 THEN CN IS CN10	IF Sand IS S100 AND Clay IS C60 THEN CN IS CN10
IF Sand IS S10 AND Clay IS C20 THEN CN IS CN90	IF Sand IS S10 AND Clay IS C70 THEN CN IS CN90
IF Sand IS S20 AND Clay IS C20 THEN CN IS CN80	IF Sand IS S20 AND Clay IS C70 THEN CN IS CN90
IF Sand IS S30 AND Clay IS C20 THEN CN IS CN80	IF Sand IS S30 AND Clay IS C70 THEN CN IS CN80
IF Sand IS S40 AND Clay IS C20 THEN CN IS CN70	IF Sand IS S40 AND Clay IS C70 THEN CN IS CN80
IF Sand IS S50 AND Clay IS C20 THEN CN IS CN70	IF Sand IS S50 AND Clay IS C70 THEN CN IS CN80
IF Sand IS S60 AND Clay IS C20 THEN CN IS CN60	IF Sand IS S60 AND Clay IS C70 THEN CN IS CN60
IF Sand IS S70 AND Clay IS C20 THEN CN IS CN50	IF Sand IS S70 AND Clay IS C70 THEN CN IS CN50
IF Sand IS S80 AND Clay IS C20 THEN CN IS CN40	IF Sand IS S80 AND Clay IS C70 THEN CN IS CN50
IF Sand IS S90 AND Clay IS C20 THEN CN IS CN20	IF Sand IS S90 AND Clay IS C70 THEN CN IS CN20
IF Sand IS S10 AND Clay IS C20 THEN CN IS CN10	IF Sand IS S10 AND Clay IS C70 THEN CN IS CN10
IF Sand IS S10 AND Clay IS C30 THEN CN IS CN80	IF Sand IS S10 AND Clay IS C80 THEN CN IS CN90
IF Sand IS S20 AND Clay IS C30 THEN CN IS CN70	IF Sand IS S20 AND Clay IS C80 THEN CN IS CN90
IF Sand IS S30 AND Clay IS C30 THEN CN IS CN70	IF Sand IS S30 AND Clay IS C80 THEN CN IS CN80
IF Sand IS S40 AND Clay IS C30 THEN CN IS CN70	IF Sand IS S40 AND Clay IS C80 THEN CN IS CN80
IF Sand IS S50 AND Clay IS C30 THEN CN IS CN70	IF Sand IS S50 AND Clay IS C80 THEN CN IS CN80
IF Sand IS S60 AND Clay IS C30 THEN CN IS CN60	IF Sand IS S60 AND Clay IS C80 THEN CN IS CN70
IF Sand IS S70 AND Clay IS C30 THEN CN IS CN50	IF Sand IS S70 AND Clay IS C80 THEN CN IS CN60
IF Sand IS S80 AND Clay IS C30 THEN CN IS CN40	IF Sand IS S80 AND Clay IS C80 THEN CN IS CN50
IF Sand IS S90 AND Clay IS C30 THEN CN IS CN10	IF Sand IS S90 AND Clay IS C80 THEN CN IS CN20
IF Sand IS S10 AND Clay IS C30 THEN CN IS CN10	IF Sand IS S100 AND Clay IS C80 THEN CN IS CN10
IF Sand IS S10 AND Clay IS C40 THEN CN IS CN80	IF Sand IS S10 AND Clay IS C90 THEN CN IS CN100
IF Sand IS S20 AND Clay IS C40 THEN CN IS CN70	IF Sand IS S20 AND Clay IS C90 THEN CN IS CN90
IF Sand IS S30 AND Clay IS C40 THEN CN IS CN70	IF Sand IS S30 AND Clay IS C90 THEN CN IS CN90
IF Sand IS S40 AND Clay IS C40 THEN CN IS CN70	IF Sand IS S40 AND Clay IS C90 THEN CN IS CN100
IF Sand IS S50 AND Clay IS C40 THEN CN IS CN70	IF Sand IS S50 AND Clay IS C90 THEN CN IS CN80
IF Sand IS S60 AND Clay IS C40 THEN CN IS CN60	IF Sand IS S60 AND Clay IS C90 THEN CN IS CN70
IF Sand IS S70 AND Clay IS C40 THEN CN IS CN50	IF Sand IS S70 AND Clay IS C90 THEN CN IS CN60
IF Sand IS S80 AND Clay IS C40 THEN CN IS CN40	IF Sand IS S80 AND Clay IS C90 THEN CN IS CN50
IF Sand IS S90 AND Clay IS C40 THEN CN IS CN10	IF Sand IS S90 AND Clay IS C90 THEN CN IS CN20
IF Sand IS S10 AND Clay IS C40 THEN CN IS CN10	IF Sand IS S100 AND Clay IS C90 THEN CN IS CN20
IF Sand IS S10 AND Clay IS C50 THEN CN IS CN80	IF Sand IS S10 AND Clay IS C100 THEN CN IS CN100
IF Sand IS S20 AND Clay IS C50 THEN CN IS CN80	IF Sand IS S20 AND Clay IS C100 THEN CN IS CN100
IF Sand IS S30 AND Clay IS C50 THEN CN IS CN70	IF Sand IS S30 AND Clay IS C100 THEN CN IS CN90
IF Sand IS S40 AND Clay IS C50 THEN CN IS CN70	IF Sand IS S40 AND Clay IS C10 THEN CN IS CN90
IF Sand IS S50 AND Clay IS C50 THEN CN IS CN70	IF Sand IS S50 AND Clay IS C10 THEN CN IS CN90
IF Sand IS S60 AND Clay IS C50 THEN CN IS CN60	IF Sand IS S60 AND Clay IS C10 THEN CN IS CN80
IF Sand IS S70 AND Clay IS C50 THEN CN IS CN50	IF Sand IS S70 AND Clay IS C10 THEN CN IS CN70
IF Sand IS S80 AND Clay IS C50 THEN CN IS CN40	IF Sand IS S80 AND Clay IS C10 THEN CN IS CN60
IF Sand IS S90 AND Clay IS C50 THEN CN IS CN10	IF Sand IS S90 AND Clay IS C10 THEN CN IS CN30
IF Sand IS S10 AND Clay IS C50 THEN CN IS CN80	IF Sand IS S80 AND Clay IS C10 THEN CN IS CN20

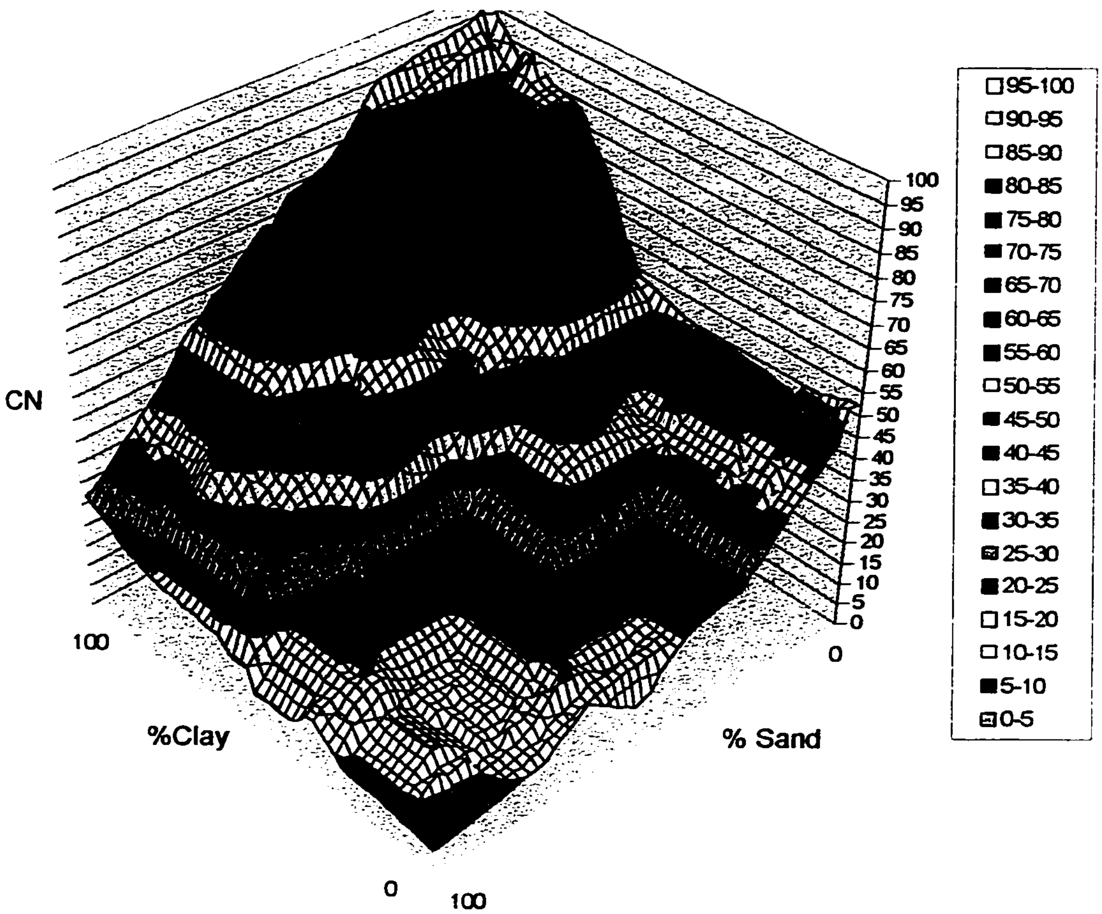


Fig 4.6 Control surface for %sand, %clay, and CN based on a rectanglar soil classification.

4.2.2 Sand Content versus Soil Moisture

Moisture content is a component in the scientific classification of natural soils and is a necessary parameter in defining important properties, such as the Atterberg limits, curve number, etc. Soil moisture condition greatly affects infiltration and thereby directly affects the watershed index. This parameter was originally defined by SCS (1972) in terms of Antecedent Moisture Condition (AMC) for calculation of a curve number in the distinct class ranges of DRY (AMCI), AVERAGE (AMCII), and WET (AMCIII). These AMC are selected and the category estimated from five-day antecedent rainfall for either the dormant season or the growing season. These adjectives have a fuzzy meaning, but in practice, this information is used to assign a deterministic value in hydrologic modeling.

In reality, moisture condition is a continuous rather than a discrete value and has additionally, a different effect on the CN index for different soil types. Based on the fuzzy meaning of the terms of the indefinite interpretation of moisture class labels used by some models (Ch.2) it was decided to increase the number of classes from three to five labels that will be presented in the final MISO model. However, at this stage the requirement was the relationship between % sand and moisture content in the form of a 10 x 10 FAM matrix. In this form the FAM matrix was solved in a double-input-single-output (DISO) fuzzy logic model, and the reaction was the curve number. The methodology of this application is the same as previously described for soil texture (Sec. 4.2.1)

The fuzzy membership functions for curve number (output), and % sand and % moisture content as inputs, are presented in Tables 4.5, 4.6, and 4.7, respectively. Fig 4.7 shows the FAM matrix for sand versus moisture content with $10 \times 10 = 100$ rules in the form of the *IF... AND... THEN.. ALSO* strategy.

The Max-Min inference was used as described in Ch.3. The GOC defuzzification method was applied to calculate the reaction to the input values as a crisp output number. The results of defuzzification are shown as a control surface diagram in Fig 4.8. A plot of the design relation between % sand, % moisture content , and curve number, derived from the control surface, is illustrated in Fig 4.9.

Table 4.5 Adjectives for Curve Number and related membership function.

Adjectives	a_1		a_2		a_3	
	Crisp	μ(x)	Crisp	μ(x)	Crisp	μ(x)
CN10		-	0.0	1.0	13.7	0.0
CN20	2.5	0.1	11.0	1.0	25.0	0.0
CN30	8.2	0.0	22.4	1.0	36.1	0.0
CN40	19.6	0.0	33.3	1.0	47.1	0.0
CN50	30.6	0.0	44.3	1.0	58.4	0.0
CN60	41.6	0.0	55.7	1.0	69.4	0.0
CN70	52.9	0.0	66.7	1.0	80.4	0.0
CN80	63.9	0.0	77.6	1.0	91.8	0.0
CN90	74.9	0.0	89.0	1.0	100	0.0
CN100	86.3	0.0	100	1.0	-	

Table 4.6 Adjectives for Moisture content and related membership function .

Adjectives	a_1		a_2		a_3	
	Crisp	μ(x)	Crisp	μ(x)	Crisp	μ(x)
M10		-	0.0	1.0	11.0	0.0
M20	0.0	0.0	11.0	1.0	22.4	0.0
M30	11.2	0.0	22.4	1.0	33.3	0.0
M40	19.6	0.0	33.3	1.0	44.3	0.0
M50	33.3	0.0	44.3	1.0	55.7	0.0
M60	44.3	0.0	55.7	1.0	66.7	0.0
M70	55.7	0.0	66.7	1.0	77.6	0.0
M80	66.7	0.0	77.6	1.0	89.8	0.0
M90	74.9	0.0	89.0	1.0	100	0.0
M100	89.8	0.0	100	1.0	-	

Table 4.7 Adjectives for% Sand and related membership function.

Adjectives	a1		a2		a3	
	Crisp	μ(x)	Crisp	μ(x)	Crisp	μ(x)
S10		-	0.0	1.0	11.0	0.0
S20	0.0	0.0	11.0	1.0	22.4	0.0
S30	11.0	0.0	22.4	1.0	33.3	0.0
S40	22.4	0.0	33.3	1.0	44.3	0.0
S50	33.3	0.0	44.3	1.0	55.7	0.0
S60	44.3	0.0	55.7	1.0	66.4	0.0
S70	55.7	0.0	66.7	1.0	77.6	0.0
S80	66.7	0.0	77.6	1.0	89.8	0.0
S90	77.9	0.0	89.0	1.0	100	0.0
S100	89.3	0.0	100	1.0	-	

M100	CN100									CN100
M90										
M80										
M70										
M60										
M50										
M40										
M30										
M20										
M10	CN90									CN10
	S10	S20	S30	S40	S50	S60	S70	S80	S90	S100

Fig 4.7 FAM matrix (10 x 10) for % Sand, % Moisture Content, and CN

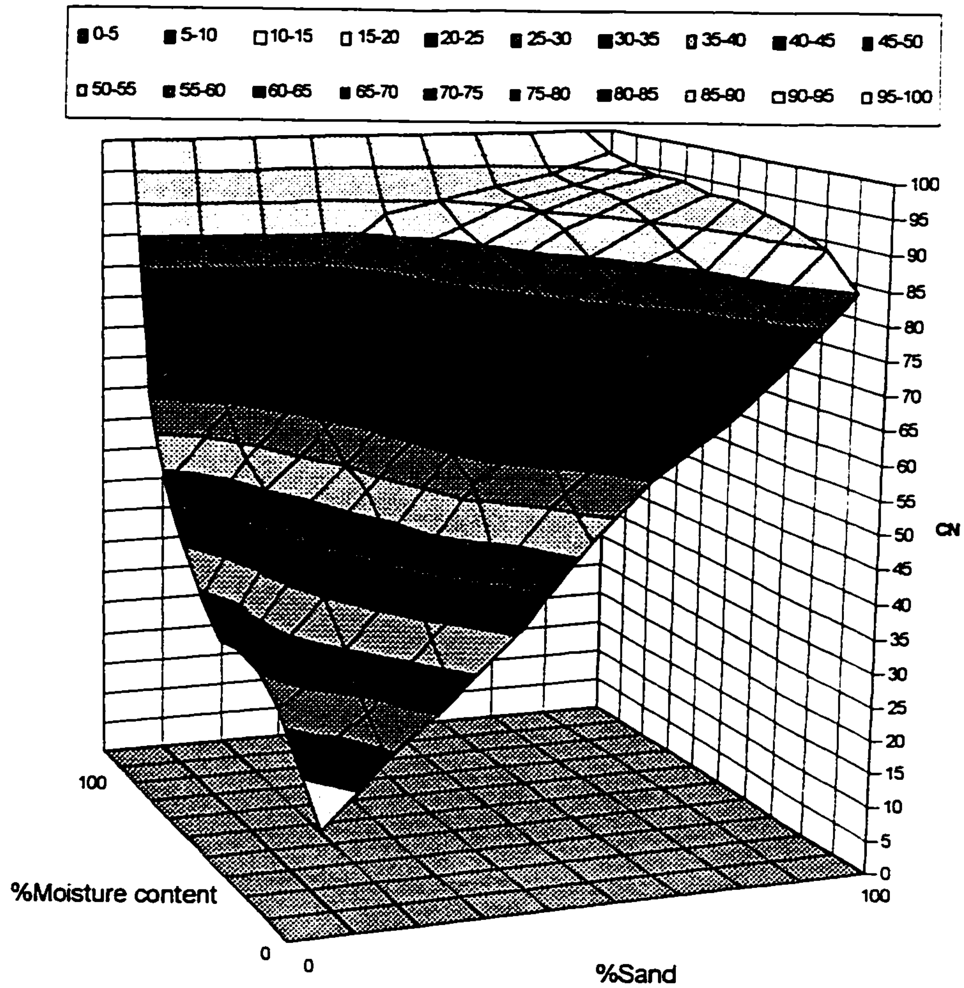


Fig 4.8 Control surface reaction of sand and moisture content for Curve Number.

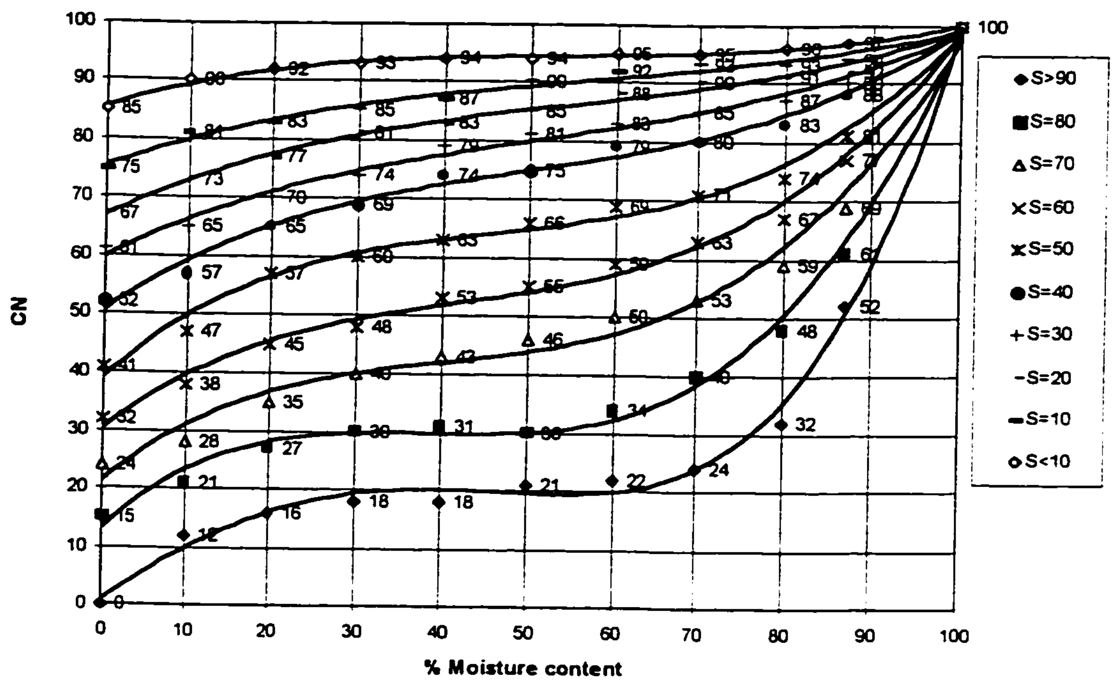


Fig 4.9 The relationship between % sand, % moisture content, and Curve Number (CN)

4.2.3 Sand Content versus Land Use

Land use is the general description for catchment cover and includes all types of vegetation, mulch and fallow as well as non-agricultural conditions such as water surfaces (lakes, swamps, etc.), urban, and suburban land use and impervious surfaces such as roads, roofs, etc. In the SCS model, land use includes a subset of qualifiers which describe the land treatment and hydrologic condition. Land treatment applies mainly to agricultural land uses and includes mechanical conservation practices such as planting in rows, contouring as against straight- row planting, and conservation tillage. Hydrologic condition is a qualitative definition of cover density and may be classified as Poor, Fair, or Good. These adjectives are based on cover density and engineering factors with fuzzy meaning.

In terms of categorizing the land use in discrete classes as from VERY HIGH runoff potential (lakes, high % impermeable area, etc.) to VERY LOW runoff potential (forest, planted forest, very high density pasture, etc.), the curve number is assumed to be a function of the kind of land use and cover density for different soils. From a global point of view, the effect of land use form having a very high density cover (forest) to very low density cover (urban area) has an increasing distribution function (IDF). As before, the procedure follows the definition of fuzzy membership function and fuzzy interval, FAM matrix, inference of rules, defuzzification, and the visualization of results. The double-input-single-output was designed for % sand and land use as inputs. The fuzzy membership function used was of the triangular shape and the fuzzy intervals for inputs and output are presented in Tables 4.8, 4.9 and 4.10 , respectively. The 10 x 10 FAM matrix of 100 rules is presented in Fig 4.10. Again, the rule base system is structured in the form of *IF...AND...THEN*. However, in this section, rules that contain the 10 label (high impervious area) unconditional strategies were applied to enforce a determined action rule. The control surface reaction to the rules system is shown in Fig 4.11. The two-dimensional plot of % sand and land use control surface is presented in Fig 4.12.

Table 4.8 Adjectives for Curve Number and related membership function.

Adjectives	a_1		a_2		a_3	
	Crisp	$\mu(x)$	Crisp	$\mu(x)$	Crisp	$\mu(x)$
CN10	-		0.0	1.0	15.0	0.0
CN20	2.5	0.1	11.0	1.0	25.0	0.0
CN30	8.2	0.0	22.4	1.0	36.1	0.0
CN40	19.6	0.0	33.3	1.0	47.1	0.0
CN50	30.6	0.0	44.3	1.0	58.4	0.0
CN60	41.6	0.0	55.7	1.0	69.4	0.0
CN70	52.9	0.0	66.7	1.0	80.4	0.0
CN80	63.9	0.0	77.6	1.0	91.8	0.0
CN90	74.9	0.0	89.0	1.0	100	0.0
CN100	86.3	0.0	100	1.0	-	

Table 4.9 Adjectives for Land Use and related membership function .

Adjectives	a_1		a_2		a_3	
	Crisp	$\mu(x)$	Crisp	$\mu(x)$	Crisp	$\mu(x)$
L10	-		0.0	1.0	11.0	0.0
L20	0.0	0.0	11.0	1.0	22.4	0.0
L30	11.2	0.0	22.4	1.0	33.3	0.0
L40	19.6	0.0	33.3	1.0	44.3	0.0
L50	33.3	0.0	44.3	1.0	55.7	0.0
L60	44.3	0.0	55.7	1.0	66.7	0.0
L70	55.7	0.0	66.7	1.0	77.6	0.0
L80	66.7	0.0	77.6	1.0	89.8	0.0
L90	74.9	0.0	89.0	1.0	100	0.0
L100	89.8	0.0	100	1.0	-	

Table 4.10 Adjectives for% Sand and related membership function.

Adjectives	a1		a2		a3	
	Crisp	$\mu(x)$	Crisp	$\mu(x)$	Crisp	$\mu(x)$
S10	-		0.0	1.0	11.0	0.0
S20	0.0	0.0	11.0	1.0	22.4	0.0
S30	11.0	0.0	22.4	1.0	33.3	0.0
S40	22.4	0.0	33.3	1.0	44.3	0.0
S50	33.3	0.0	44.3	1.0	55.7	0.0
S60	44.3	0.0	55.7	1.0	66.4	0.0
S70	55.7	0.0	66.7	1.0	77.6	0.0
S80	66.7	0.0	77.6	1.0	89.8	0.0
S90	77.9	0.0	89.0	1.0	100	0.0
S100	89.3	0.0	100	1.0	-	

L100	**CN100**									**CN100**
L90										
L80										
L70										
L60										
L50					**CN60**					
L40										
L30										
L20										
L10	**CN80**									**CN10**
	S10	S20	S30	S40	S50	S60	S70	S80	S90	S100

Fig 4.10 The FAM matrix [10 x 10] for % Sand and Land use

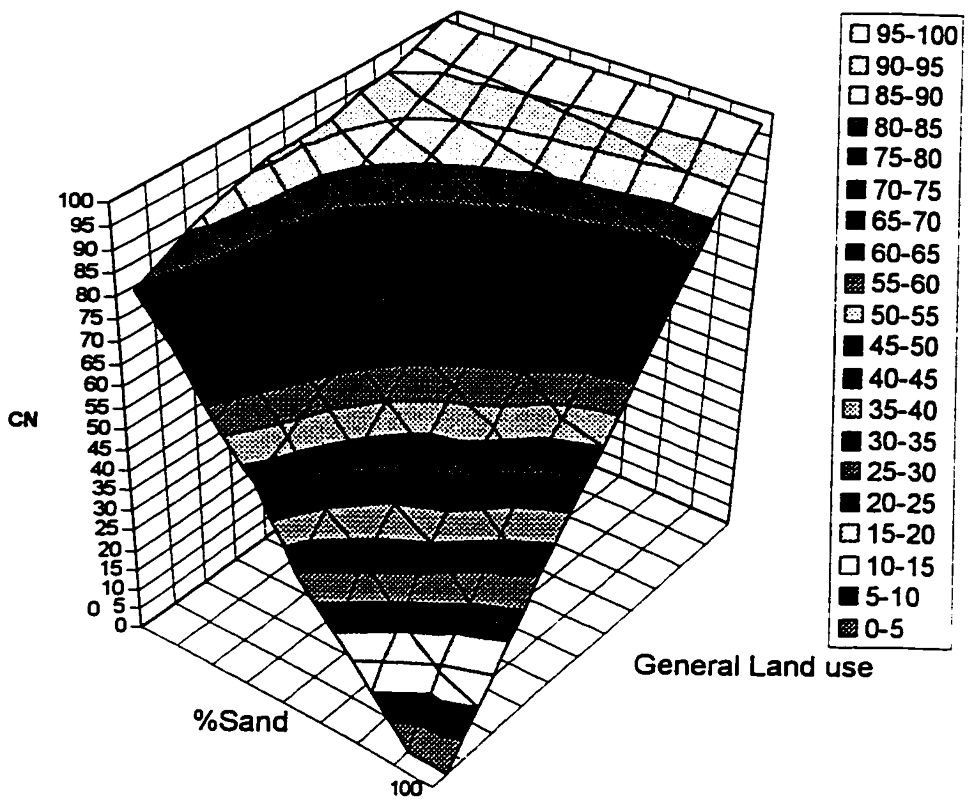


Fig 4.11 Control surface of reaction of % sand and land use to Curve Number

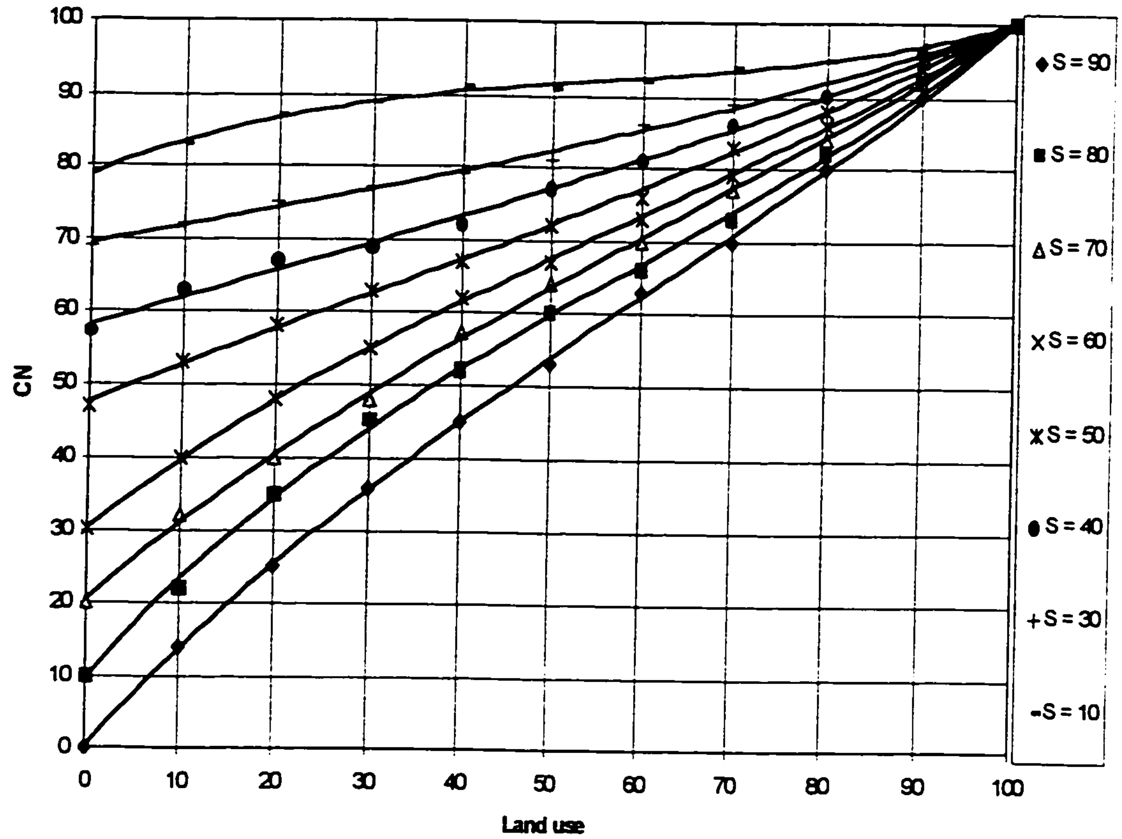


Fig 4.12 Relation between % sand, land use and Curve Number.

4.2.4 Sand Content versus % Impermeable Area

Urbanization of a watershed is a situation in which impervious surfaces cover, or will soon cover, a considerable portion of the area. Impervious surfaces include roads, sidewalks, parking lots and buildings. Most urban areas are only partially covered with impervious surfaces: and permeable surfaces remain an important factor in runoff estimates. Moreover, permeable land surface in urban areas mostly differ from those in agricultural lands. Permeable land surfaces in urban areas are more disturbed and often evidence mixing with imported material.

Assuming infiltration is influenced more by soil texture (% sand) and % impermeable area in urban and sub-urban areas, the curve number was derived as a function of % sand and % impermeable area.

The fuzzy membership function, fuzzy intervals and rule-based system were developed as described for the previous DISO models. Again a DISO FLC model was used and the fuzzy membership function used was of triangular shape. Ten fuzzy intervals of input values were used. The adjectives and related membership functions are presented in Tables 4.11, 4.12 and 4.13 for output and inputs respectively. The fuzzy membership functions are shown in Fig 4.13 and the FAM surface for % sand and % impermeable area is presented in Fig 4.14. One hundred rules were used in the knowledge-base system in the form of an IF...AND..THEN..ALSO.. strategy and a Min-Max inference was applied to evaluate the rules-based system. The translator of fuzzy output to crisp output utilized the COG defuzzification method.

In this component, in addition to conventional fuzzy logic based on conditional relations between a rule's antecedent and consequent, situations occur in which some rules have more power. In some cases a rule will have absolute power to determine its consequent. This unconditional situation will be elaborated on later in describing the MISO model. The result of the reaction control surface to inputs and a plot of the relationship between % sand and % impermeable area is shown in Fig 4.15. The results of this section are discussed at the end of Ch.5.

Table 4.11 Adjectives for Curve Number and related membership function.

Adjectives	a_1		a_2		a_3	
	Crisp	μ(x)	Crisp	μ(x)	Crisp	μ(x)
CN10		-	0.0	1.0	15.0	0.0
CN20	2.5	0.1	11.0	1.0	25.0	0.0
CN30	8.2	0.0	22.4	1.0	36.1	0.0
CN40	19.6	0.0	33.3	1.0	47.1	0.0
CN50	30.6	0.0	44.3	1.0	58.4	0.0
CN60	41.6	0.0	55.7	1.0	69.4	0.0
CN70	52.9	0.0	66.7	1.0	80.4	0.0
CN80	63.9	0.0	77.6	1.0	91.8	0.0
CN90	74.9	0.0	89.0	1.0	100	0.0
CN100	86.3	0.0	100	1.0	-	

Table 4.12 Adjectives for Impermeable area and related membership function .

Adjectives	a_1		a_2		a_3	
	Crisp	μ(x)	Crisp	μ(x)	Crisp	μ(x)
IMP10	-	0.0	1.0	15.0	0.0	
IMP20	0.0	0.0	11.0	1.0	25.0	0.0
IMP30	11.2	0.0	25.0	1.0	34.0	0.0
IMP40	19.6	0.0	34.0	1.0	45.0	0.0
IMP50	33.3	0.0	45.0	1.0	55.0	0.0
IMP60	44.3	0.0	55.0	1.0	66.0	0.0
IMP70	55.7	0.0	66.0	1.0	77.0	0.0
IMP80	66.7	0.0	77.0	1.0	90.0	0.0
IMP90	74.9	0.0	90.0	1.0	100	0.0
IMP100	90.0	0.0	100	1.0	-	

Table 4.13 Adjectives for% Sand and related membership function.

Adjectives	a1		a2		a3	
	Crisp	μ(x)	Crisp	μ(x)	Crisp	μ(x)
S10		-	0.0	1.0	11.0	0.0
S20	0.0	0.0	11.0	1.0	22.0	0.0
S30	11.0	0.0	22.0	1.0	33.0	0.0
S40	22.0	0.0	33.0	1.0	44.0	0.0
S50	33.0	0.0	44.0	1.0	55.0	0.0
S60	44.3	0.0	55.0	1.0	66.0	0.0
S70	55.0	0.0	66.0	1.0	77.0	0.0
S80	66.0	0.0	77.0	1.0	89.0	0.0
S90	77.0	0.0	89.0	1.0	100	0.0
S100	89.0	0.0	100	1.0	-	

IMP100	**CN100**	**CN100**	**CN100**	**CN100**	**CN100**	**CN100**	**CN100**	**CN100**	**CN100**	**CN100**
IMP90										
IMP80										
IMP70										
IMP60										
IMP50					**CN50**					
IMP40										
IMP30										
IMP20										
IMP10	**CN80**									**CN10**
	S10	S20	S30	S40	S50	S60	S70	S80	S90	S100

Fig 4.13 The FAM matrix [10 x 10] for % sand and % impermeable area

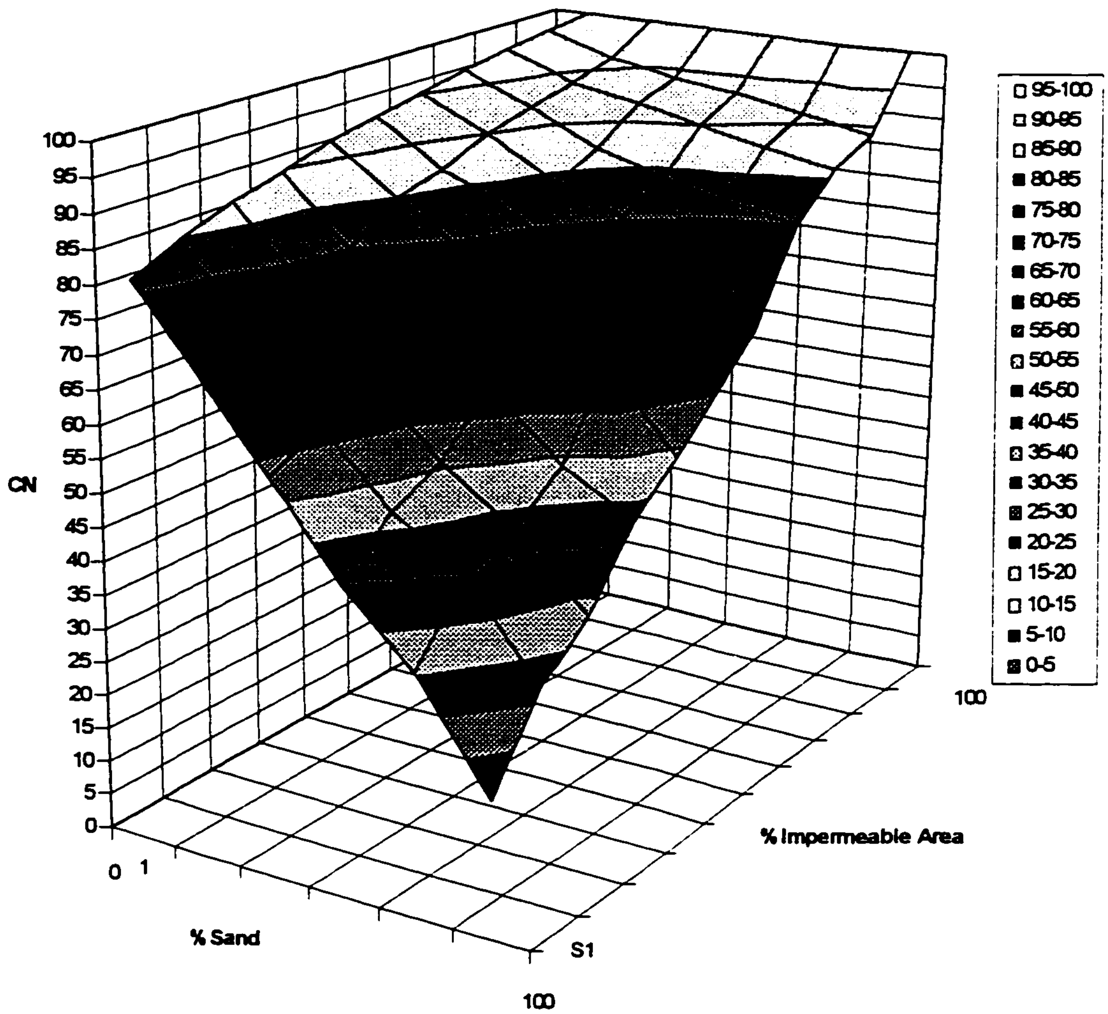


Fig 4 .14 Control surface for Curve Number as a function of % sand and % impermeable area for urban areas.

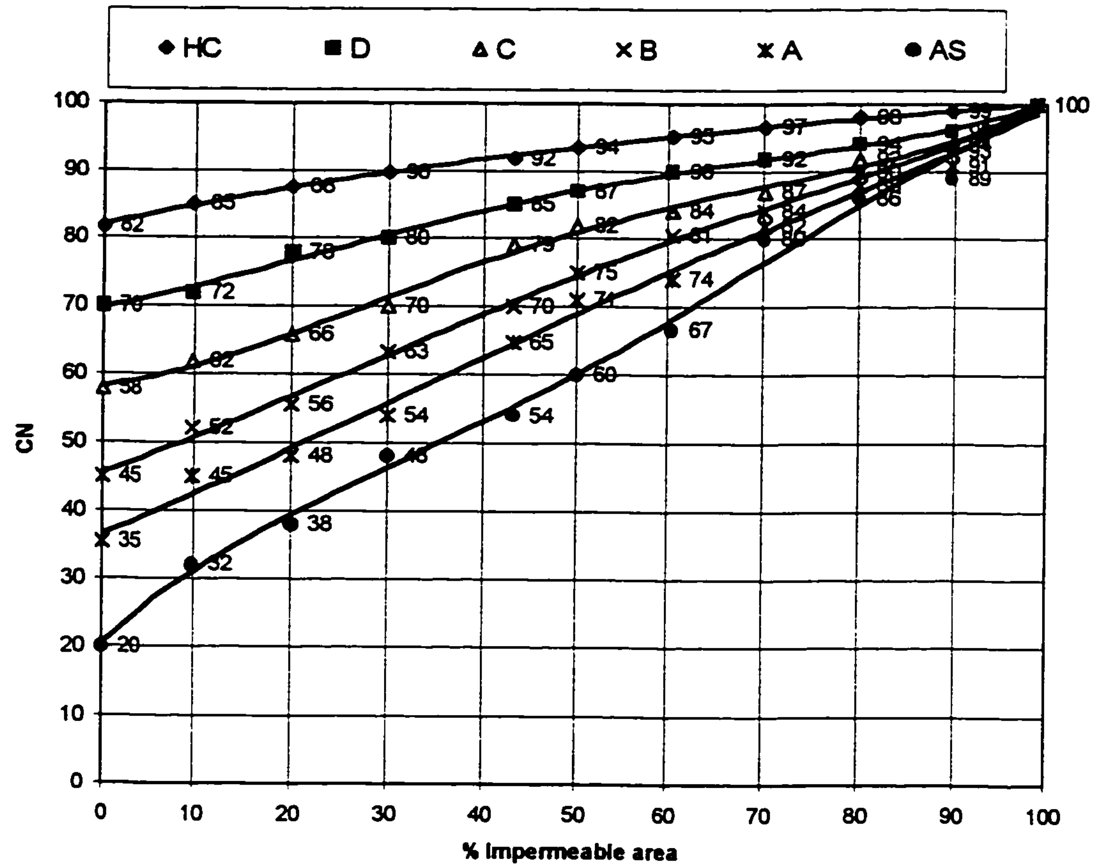


Fig 4.15 Relation between % sand, Impermeable area and Curve Number for urban areas.
Note: AS, A, B, C,D , and HC are represenred for soil texture as sandy soil to heavy clay.

5. MULTI-INPUT-SINGLE-OUTPUT (MISO) FUZZY LOGIC MODEL

5.1 General

Multivariable fuzzy control systems have received a great deal of recent attention from engineers and scientists. This interest results from a recognition that:

(i) Real control systems are multidimensional (multi-inputs), and

(ii) Computer implementation of real physical systems requires the processing of a large data base, which is often accompanied by memory overload.

Analysis and design procedures for each system are consequently very difficult. Shakouri et al. (1982) proposed a fuzzy control algorithm for multivariable systems which is based on a state space model of the system. Walichiewicz (1984) proposed a multidimensional fuzzy controller using the decomposition of rules through intersection coefficients. Gupta et al. (1986) developed a multivariable fuzzy control for an open system. Koczy and Hirota (1992, 1993) argued that decomposition of rules through intersection produced better sensitivity in approximation reasoning and low memory requirement. Gegov et al. (1994), Gegov and Frank 1996), and Valente de Oliveira (1995) considered a decentralized method for solving a multi-variable system by passive and active decomposition methods using off-line and up-line algorithms. They found that a decentralized method was suitable under certain conditions. The use of a Genetic Algorithm (GA) in a hierarchical multi-variable fuzzy controller (HIFLC) to reduce the number of rules was investigated by Linkens and Nyongesa (1996). They assumed that the number of rules is a linear function of the number of state inputs with a conditional rule inference.

5.2 MISO Model

In this chapter a multi-variable fuzzy logic controller, based on the theory presented in Ch.3 for an open-loop control system by parallel rule firing, is presented. Open-loop fuzzy systems are those systems, described by fuzzy relationships, in which the outputs have no effect upon the input control actions.

This comprises a division of the crisp values of variables such as % cover density, % sand, % moisture condition and % impermeable area into classes and the assigning of a fuzzy label in the universe of discourse (which ranges from 0 to 100 for all inputs and outputs in this study) to each of the classes of crisp inputs. As an example, soil moisture condition class labels may comprise VERY DRY, DRY, MOIST, FC, and, SATURATION. The membership functions defined on the input variables are applied to their actual values to determine the degree of truth for each rule premise. The definition of domain and universe of discourse for a fuzzy logic controller is very important. These definitions are normally based on human expert experience; however, at times the developer has to determine these mathematically.

As described in the previous section, a triangular distribution (Eq. 3.8) was used to define the fuzzy membership function for the soil texture , the cover density, the moisture condition, and the % impermeable area as inputs, and for the curve number (CN) in the output space. However, since in this model the fuzzy quantity space was modeled by use of linguistic terms which are used in the rules, the fuzzy relation degenerates to a conventional binary relation, with the ranges of the input spaces ($X^1 \ldots X^n$) based on the theory presented in Ch.3.

The specific MISO model developed in this study is named $FLEW^{CN}$ (Fuzzy Logic Expert Watershed model for Curve Number),

5.2.1 Effect of Membership Function on Inference

In this section, a set of rational properties for membership functions involved in an inference is postulated. Basically, the nature of these properties is semantic and related to information processing issues. It should be stressed that other properties may be considered in the design of interface membership functions. Two main properties are as set out below:

(i) Semantic concern. Semantics of relevance to the interface comprise:

(a) The number of adjective terms (or referential fuzzy set) used. Human beings only can memorize and utilize a limited number of linguistic terms per fuzzy variable. Therefore, this number should not exceed a practical limit of 7 ± 2 different terms.

(b) The "natural zero" positioning: one point should exist to represent the natural zero position. That is, whenever required, there should be a linguistic term that can include zero.

(c) Coverage of the universe of the discourse (UoD): the linguistic term should cover the entire universe of discourse so that every datum value has a linguistic representation.

(d) Normalization of the membership functions: since each membership function represents a linguistic term with a clear semantic interpretation, then at least one datum in the universe of discourse (UoD) should exhibit full matching with each membership function.

(e) Distinguishability of the linguistic terms: the adjectives should have a clear semantic meaning; i.e. they should be distinct from each other.

(ii) Information processing concerns: Since inference can also be viewed as the pre/postprocessing blocks of a fuzzy system, from an information point of view the interaction between the membership functions of the referential fuzzy set and an interface should:

(a) Allow information equivalence between the original and the converted data i.e. an inference should conserve the information inputted into the processing block. This is directly related to the necessity of ensuring that the overall system will process the input signals and not something which is unrepresentative.

(b) Contribute to improving the overall system capabilities. This includes both the level and the optimization of explicit performance.

(c) Keep computational requirement as low as possible. This includes both memory requirement and time of execution.

5.3 Organization of Input/Output

A conventional computer algorithm cannot handle ambiguities. On the other hand, programs based on fuzzy theory are specifically designed for ambiguities. Given a fuzzy set of the possible conditions in the real world, representation of an ambiguity simply means that more than one member has a non-zero confidence interval.

The selection of the most appropriate shape for a membership function depends on its intended use. If the final output is to be non-numeric, then a flat-topped membership function with adjacent functions having maximum overlap are usually best. However, if the output is to be numeric, then peaked functions intersecting at one half confidence intervals are usually the most appropriate.

In terms of computer programming, the method of representation of a fuzzy membership function in the input and output spaces for this study is in matrix format, as below:

(i) Input space. This is a four dimensional space for which the dimensions are sand, cover density, moisture condition, and impermeable area defined as:

$$X^{(M)} = \left| X^{(1)} \;\; X^{(2)} \;\; X^{(3)} \;\; X^{(4)} \right| \tag{5.1}$$

where $X^{(1)}$, $X^{(2)}$,$X^{(3)}$, and $X^{(4)}$ are sand, cover density, moisture content, and impermeable area, respectively. The units of the input space are percentage of interval space of the fuzzy sets. The effect of each variable on the curve number is based on a logical relationship between inputs and output. For instance, the behavior of soil texture is based on runoff potential which ranges from very low runoff potential (very sandy soil) to very high runoff potential (very low sand or very high clay soil). Each variable in the input space is fuzzified into five membership functions as below:

$$X^{(1)} = \begin{vmatrix} VLS = X_1^{(1)} \\ LS = X_2^{(1)} \\ MS = X_3^{(1)} \\ VLS = X_4^{(1)} \\ VHS = X_5^{(1)} \end{vmatrix} \quad X^{(2)} = \begin{vmatrix} VLC = X_1^{(2)} \\ LC = X_2^{(2)} \\ MC = X_3^{(2)} \\ HC = X_4^{(2)} \\ VHC = X_5^{(2)} \end{vmatrix} \quad X^{(3)} = \begin{vmatrix} VDRY = X_1^{(3)} \\ DRY = X_2^{(3)} \\ MOIST = X_3^{(3)} \\ FC = X_4^{(3)} \\ SATU = X_5^{(3)} \end{vmatrix} \quad X^{(4)} = \begin{vmatrix} VLIA = X_1^{(4)} \\ LIA = X_2^{(4)} \\ MIA = X_3^{(4)} \\ HIA = X_4^{(4)} \\ VHIA = X_5^{(4)} \end{vmatrix} \tag{5.2}$$

The fuzzy sets were arranged in a 2-D matrix form for simplicity of programing. Each membership function was presented as a vector in [0, 1] space. The $X^{(M)}$ input space relation matrices were then as below:

(a) For sand,

$$X^{(1)} = \begin{array}{c|ccccccc|} & VLS & VLS & LS & MS & HS & VHS & VHS \\ VLS & 1.0 & 1.0 & 0 & 0 & 0 & 0 & 0 \\ LS & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ MS & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 \\ HS & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 \\ VHS & 0 & 0 & 0 & 0 & 0 & 1.0 & 1.0 \end{array} \quad (5.3)$$

(b) For cover density:

$$X^{(2)} = \begin{array}{c|ccccccc|} & VLC & VLC & LC & MC & HC & VHC & VHC \\ VLC & 1.0 & 1.0 & 0 & 0 & 0 & 0 & 0 \\ LC & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ MC & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 \\ HC & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 \\ VHC & 0 & 0 & 0 & 0 & 0 & 1.0 & 1.0 \end{array} \quad (5.4)$$

(c) For moisture content:

$$X^{(3)} = \begin{array}{c|ccccccc|} & VDRY & VDRY & DRY & MOIS & FC & SATU & SATU \\ VDRY & 1.0 & 1.0 & 0 & 0 & 0 & 0 & 0 \\ DRY & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ MOIS & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 \\ FC & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 \\ SATU0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 1.0 \end{array} \quad (5.5)$$

(d) For impermeable area:

$$X^{(4)} = \begin{array}{c|ccccccc|} & VLIA & VLIA & LIA & MIA & HIA & VHIA & VHIA \\ VLIA & 1.0 & 1.0 & 0 & 0 & 0 & 0 & 0 \\ LIA & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ MIA & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 \\ HIA & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 \\ VHIA & 0 & 0 & 0 & 0 & 0 & 1.0 & 1.0 \end{array} \qquad (5.6)$$

(ii) Output space. The curve number (CN) is the only output of this model and is presented in fuzzified form in the output space. In terms of its definition CN ranges from 0 to 100. In other words, the UoD for the output space is [0, 100]. The crisp values of the output space [O] are fuzzifed in five intervals and the membership function is:

$$O = \begin{vmatrix} VLCN = O_1 \\ LCN = O_2 \\ MCN = O_3 \\ HCN = O_4 \\ VHCN = O_5 \end{vmatrix} \qquad (5.7)$$

The relation matrix of the output space is:

$$[O] = \begin{array}{c|ccccccc|} & VLCN & VLCN & LCN & MCN & HCN & VHCN & VHCN \\ VLCN & 1.0 & 1.0 & 0 & 0 & 0 & 0 & 0 \\ LCN & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ MCN & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 \\ HCN & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 \\ VHCN & 0 & 0 & 0 & 0 & 0 & 1.0 & 1.0 \end{array} \qquad (5.8)$$

5.4 Program Structure and Strategy

In an expert system reasoning follows one of two strategies: forward chaining or backward chaining. In principle, forward chaining draws all possible consequences from rules in light of available data-whereas backward chaining selects rules appropriate for achieving specific goals. Fuzzy control is a branch of fuzzy expert systems which has a fairly narrow purpose, process control, and is a highly developed field. A fuzzy algorithm generally is a forward chaining strategy in which data are combined with rules to deduce new information. From a programming point of view, fuzzy logic is a mixed algorithm with logical programing style as in an expert system development, because fuzzy logic control has an *IF..AND..THEN...ALSO* knowledge-based system. However, in addition, fuzzy logic is also a procedural process. Numerical (N) data or crisp values are transferred to fuzzy data or linguistic (L) "adjectives" (N → L) for fuzzification. Inference is a computational process which calculates the fuzzy relations based on the rule-based system as for an expert system. However, the (N→L) is followed by a transfer back to numerical values in the defuzzification process (L→N). In the case of multi-input-single-output (MISO) models and the multi-input-multi-output (MIMO) models there are several approaches for the design of I/O.

As mentioned previously, in this study some unconditional rules occur (e.g. when there is a very high moisture content and a very large percentage of impermeable area) which can reduce the number of rules. A problem occurred when the fuzzy knowledge builder (FKB) software Fuzzy Knowledge Builder V.2.5 (McNeill 1994) was used to generate a rule-based system for an input space of four variables with dimensions of (5*5*5*8), in that eighteen hundred rules were generated as a black box and this was very difficult to manipulate in a low RAM computer.

The parallel algorithm with decomposition was used to manipulate the multi-inputs fuzzy model in this study using the mathematical model for MISO systems described in Ch.3. The concept used to connect the I/O and transform the flow of numerical input to numerical output is illustrated in Fig 5.1. The overall procedure of the fuzzy model was coded in C language to build the steps of fuzzification, rules evaluation, inference engine, and

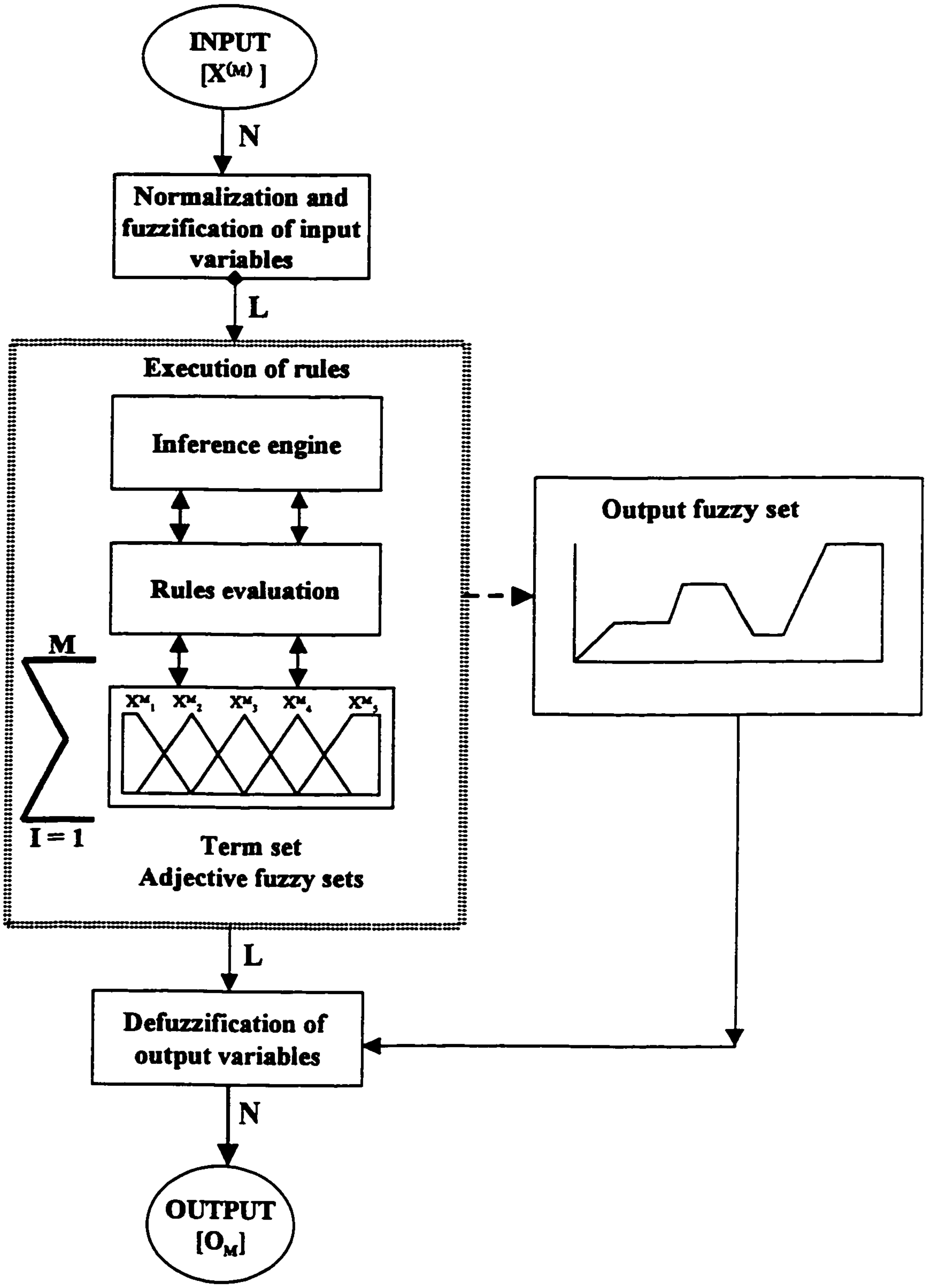


Fig 5. 1 Transformation of input to output for numerical (N) and linguistic (L) variables for fuzzy logic control process.

defuzzification process. The model comprised three algorithm A, B, and C listed below. Algorithm A initialized the overall model and algorithm B connected the inference engine to the user interface. Algorithm C initialized the keys to switch algorithm B to the user interface.

Algorithm A

Step 1: Collect N samples of numerical data from the universe of discourse (UoD). The collection of data should be well distributed over the UoD.

Step 2: Select the number of variables and their membership functions (n) with $n > 1$. The value of n should kept small ($3 \le n \le 5$). An increase in n is only required when higher levels of nonlinear processing at the interface must be enforced (then n can be chosen as $5 \le n \le 10$).

Step 3: Select the shape of the membership functions. Membership functions with three parameters, such as departure point, center point and aperture point are sufficient for most applications (Eqs. 5.3. to 5.7)

Step 4: Calculate the membership values based on Step 3, minimum operators (Eq. 3.2) and maximum operators (Eq. 3.3).

Step 5: Interpolate the parameters in Step 3 for membership functions based on linear interpolation.

Step 6: Get the Max operators on the fuzzy set $X^N{}_M$ based on Step 3 and 5.

Step 7: Run inference engine by solving Eq. 3.39 with regard to Eqs. 3.11a and 3. 11b for the MISO.

Step 8: Defuzzify fuzzy inputs to crisp output by solving Eqs. 3.16 and 3.17 based on the COG method and singleton membership function.

Step 9: Generate defuzzified output data in visual form using an interface program.

Algorithm B

Step 1: Define the UoD $[X^M]$ (Eq. 5.1) and membership function $[X^M{}_N]$ (Eq. 5.2).

Step 2: Solve Algorithm A for Input/Output space.

Step 3: Define the membership shape and solve Eqs. 3.7, 3.8 and 3.9 as applicable.

Step 4: Define the graph (mode and driver).

Step 5: Define position (x, y) for each graph in terms of size and memberships.

Step 6: Get Step 1 to initialize singleton of terms [$X^{M}{}_{N}$] and the UoD [X^{M}].

Step 7: Calculate the offset for the singleton Xi = {x [N_SET - 1]-X[0]/100.0} for $X_{singleton}$ = (xo - X[0]/Xi)

Step 8: Define the controller key action.

Step 9: Repeat Step 6 for all the UoD.

Algorithm C

Step 1: Draw the membership function graph, with respect to algorithms A and B.

Step 2: Switch action key to Step 8 in Algorithm B using an open-loop strategy.

Step 3: Visualize all on-screen based on defined Algorithm B.

Step 4: Do action with defined input/output with user interface.

Step 5: Repeat Steps 5, 6 and 7 of Algorithm B.

5.5 Inference Engine in a MISO

An inference mechanism is a procedure by which a control action can be inferred given a rule-base and a set of input values. The main difficulty in using a fuzzy rule-base control or fuzzy expert system is the large number of rules required if the number of state variables is not very small. Both memory space and time increase exponentially in modeling using (fuzzy) rules. A highly complex computational problem is, therefore, limited in applicability as can be noticed when analyzing an existing fuzzy rule-based system in which the number of variables is severely restricted. A partial solution to this problem is offered by the rules interpolation method that reduces the number of terms per variable. However, the exponent of the number of rules does not change but the use of rule interpolation itself reduces considerably the size and processing time of the rule-base using an alpha-cut and Lagrangian interpolation method (when rules have an equal weight). These problems become more evident when dealing with unequal rule power or unconditional rules. In a conventional rule-based system all rules are treated with equal rule weight. In this model, for instance, when the moisture content is close to saturation, or the percentage of impermeable area is very high, then the curve number value should be close to 100. This means that in reality the rules of the saturation membership function have absolute power on the rule's consequent. For this specific case, a method was proposed to reduced the number of rules by giving power to such rules that they alone act to determinate the final consequent of all the rules defuzzified to produce crisp output. Smith (1993) proposed a situation-specific switching to a different defuzzification method. In practice, the selection of an appropriate defuzzification method depends on application specifics such as static, dynamic, statistical and implementation properties (Runkler 1997). However, this kind of selection requires considerable implementation effort. Therefore, this study was restricted to a fuzzy rule-based system with a fixed defuzzification method. As for the DISO model (used in Ch.4) the COG standard defuzzification method was chosen. Also, fuzzy rule-based systems not only depend on the defuzzification method used but also on the inference and composition operators selected. It is important to consider the compatibility of all operators used in a fuzzy rule-based system. The Min-Max and Max-Prod methods (Eqs. 3.11a and 3.11b) are the most popular general form used in the inference engine of a MISO. Additionally, Eq. 3.39 is commonly used.

The process of generating the rules for a MISO model with four inputs and one output is presented in Fig 5.2. As seen, the number of rules for this model, without a reduction in rules, increases exponentially with the number of membership functions [number of rules = $M^N = 5^4 = 625$, where N is the number of variables and M is the number of terms]. Also assuming the input space is defined by Eqs. 5.1 to 5.6, that output space is given by Eq. 5.7 or 5.8 and that only one process is considered (one crisp input for each variable and with a response of one crisp output value), then based on Eq 3.33, the possible number of rules generated for this example are presented in Table 5.1. The processes of fuzzification, rules evaluation and defuzzification are illustrated in Fig 5.3.

In this example, only one process was considered, because of the complexity mentioned before. As shown in Fig 5.3, the eight membership functions for the input space (X^M) are: X^1_1, X^1_2, X^2_1, X^2_2, X^3_1, X^3_2, X^4_4, X^4_5 with O4 and O5 in the output space [Z]. These input

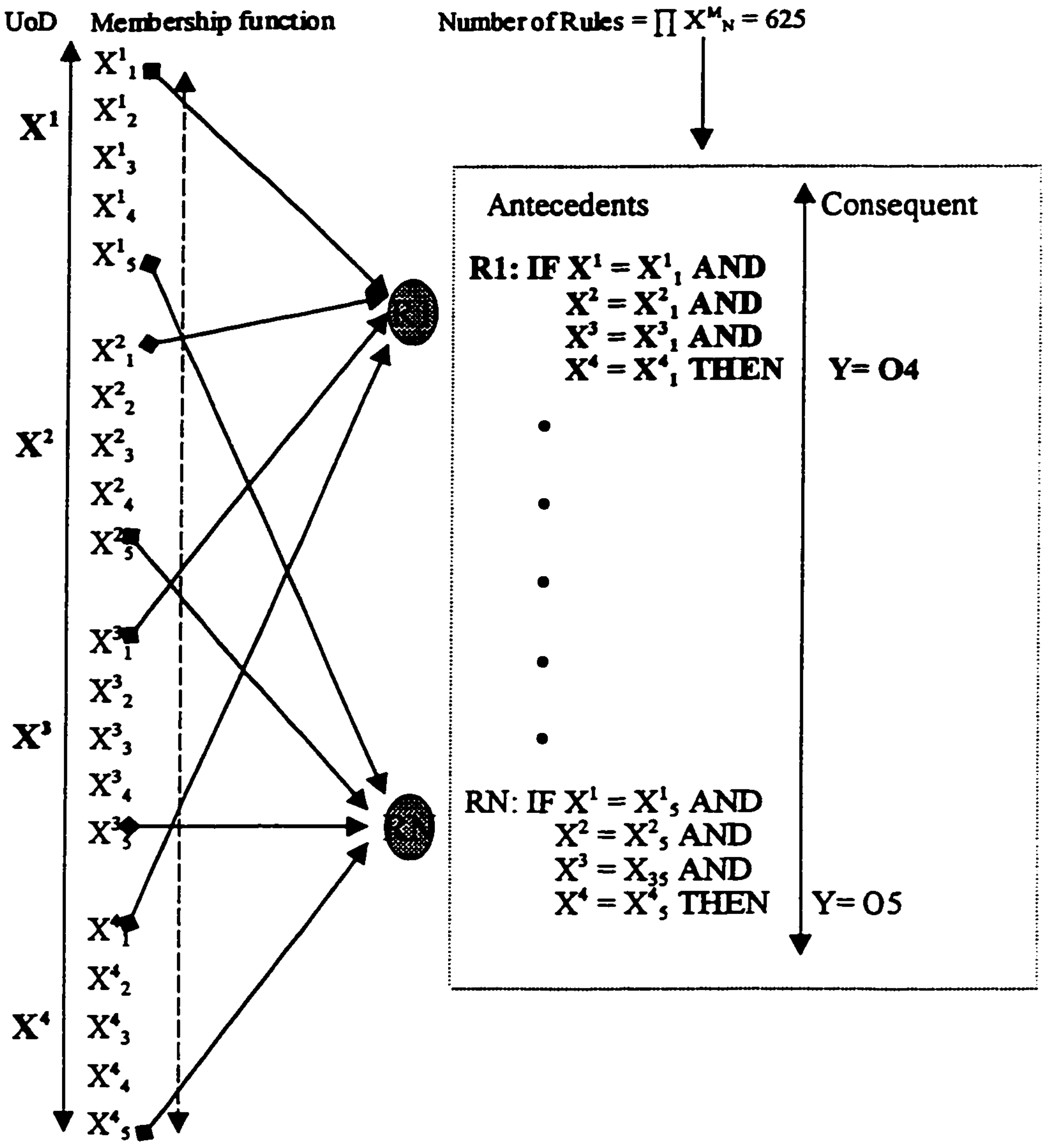


Fig 5.2 Processes of rules generation for a MISO fuzzy logic model

Table 5.1 Number of rules generated in Fig 5.2.

R1:	**IF $X^1 = X^1_1$ AND $X^2 = X^2_1$ AND $X^3 = X^3_1$ AND $X^4 = X^4_4$**	**THEN Z = O4**	**ALSO**
R2:	**IF $X^1 = X^1_1$ AND $X^2 = X^2_1$ AND $X^3 = X^3_2$ AND $X^4 = X^4_4$**	**THEN Z = O4**	**ALSO**
R3:	**IF $X^1 = X^1_2$ AND $X^2 = X^2_2$ AND $X^3 = X^3_1$ AND $X^4 = X^4_5$**	**THEN Z = O5**	**ALSO**
R4:	**IF $X^1 = X^1_2$ AND $X^2 = X^2_2$ AND $X^3 = X^3_2$ AND $X^4 = X^4_5$**	**THEN Z = O5**	**ALSO**
R5:	**IF $X^1 = X^1_1$ AND $X^2 = X^2_2$ AND $X^3 = X^3_1$ AND $X^4 = X^4_4$**	**THEN Z = O4**	**ALSO**
R6:	**IF $X^1 = X^1_1$ AND $X^2 = X^2_2$ AND $X^3 = X^3_2$ AND $X^4 = X^4_4$**	**THEN Z = O4**	**ALSO**
R7:	**IF $X^1 = X^1_1$ AND $X^2 = X^2_2$ AND $X^3 = X^3_1$ AND $X^4 = X^4_5$**	**THEN Z = O5**	**ALSO**
R8:	**IF $X^1 = X^1_1$ AND $X^2 = X^2_2$ AND $X^3 = X^3_2$ AND $X^4 = X^4_5$**	**THEN Z = O5**	**ALSO**
R9:	**IF $X^1 = X^1_2$ AND $X^2 = X^2_1$ AND $X^3 = X^3_1$ AND $X^4 = X^4_4$**	**THEN Z = O4**	**ALSO**
R10:	**IF $X^1 = X^1_2$ AND $X^2 = X^2_1$ AND $X^3 = X^3_2$ AND $X^4 = X^4_4$**	**THEN Z = O4**	**ALSO**
R11:	**IF $X^1 = X^1_2$ AND $X^2 = X^2_2$ AND $X^3 = X^3_1$ AND $X^4 = X^4_5$**	**THEN Z = O5**	**ALSO**
R12:	**IF $X^1 = X^1_2$ AND $X^2 = X^2_2$ AND $X^3 = X^3_2$ AND $X^4 = X^4_5$**	**THEN Z = O5**	**ALSO**
R13:	**IF $X^1 = X^1_2$ AND $X^2 = X^2_2$ AND $X^3 = X^3_1$ AND $X^4 = X^4_4$**	**THEN Z = O4**	**ALSO**
R14:	**IF $X^1 = X^1_2$ AND $X^2 = X^2_2$ AND $X^3 = X^3_2$ AND $X^4 = X^4_4$**	**THEN Z = O4**	**ALSO**
R15:	**IF $X^1 = X^1_2$ AND $X^2 = X^2_2$ AND $X^3 = X^3_1$ AND $X^4 = X^4_5$**	**THEN Z = O5**	**ALSO**
R16:	**IF $X^1 = X^1_2$ AND $X^2 = X^2_2$ AND $X^3 = X^3_2$ AND $X^4 = X^4_5$**	**THEN Z = O5**	**ALSO**

Ri = Rule number

X^M = Input number for i=1,...4.

X^M_N = Membership function for N = 1,...5.

Z = output space with membership function Oj for j =1,...5.

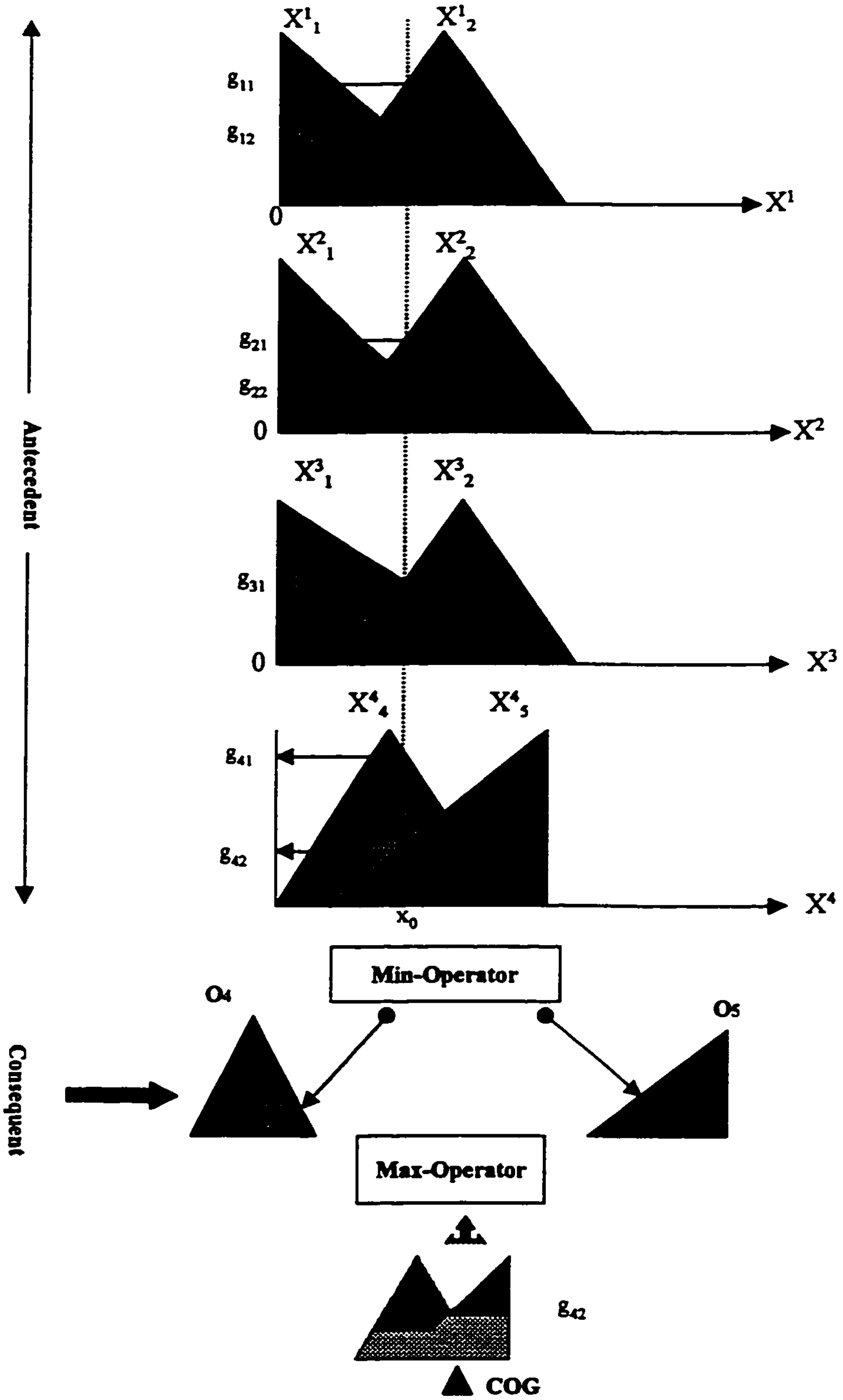


Fig 5.3 Fuzzy inference schematic for MISO model for generated rules presented in Table 5.2.

membership functions are overlapped with 50% partial similarity of degree of truth. The first step in inference is to obtain the value of each membership function. This gives the grade (Eq. 3.8) for each variable obtained by calculating the value of the function for a given input x_o (e.g., the degree of acceptance for variable $X^1 = [X^1_1 = g_{11}, X^1_2 = g_{12}]$). The same procedure is performed for all functions. For the semantics of the AND or ALSO operator the minimum operator (such as Eq 3.2) is applied to the functions in the antecedent section by selecting the minimum grade. The consequent processing in Table 5.1 involves obtaining the functions O4 and O5 by lopping off the part greater than the value of the membership function that can affect the output. As shown in Fig 5.3, the minimum grade (g) belongs to variable X^1 with membership $X^1_1=g_{12}$ and X^4 with membership $X^4_5 = g_{42}$ As may be noted the output space [Z] is influenced by variable X^4 rather than the other variables. The output space Z (CN) therefore is affected more by a high impermeability rather than soil, cover or even low moisture content (except the saturation level). In this way it is inferred that the variables X^3 and X^4 are unequal to other variables in order to reduce the number of rules without affecting the results. In any combination of rule generation the number of rules is reduced (from 16 to 4 in this example) by similarity properties. The relationship between each input space (antecedent) and output space (consequent) is illustrated in Fig 5.4. As shown, it is clear that the variables X^3 and X^4 (with membership function X^4_5 and X^3_5) have a maximum effect on the rules' consequent. It is therefore not necessary to have different antecedents and degree of acceptance for which the determination of the output space will be the same. In application this results from the number of rules fired.

5.5.1 Rule Firing Method

Fuzzy control and fuzzy reasoning are similar, the difference being that fuzzy control is a specialized subset of fuzzy reasoning. In both, there are two types of inference engine (rule evaluation) which enable emulation of two different types of thinking; deductive logic (serial rule firing) and inductive logic (parallel rule firing). Which type is better depends partly on the problem, partly on how the input data are acquired and partly on the preferences of the domain expert and the knowledge engineer.

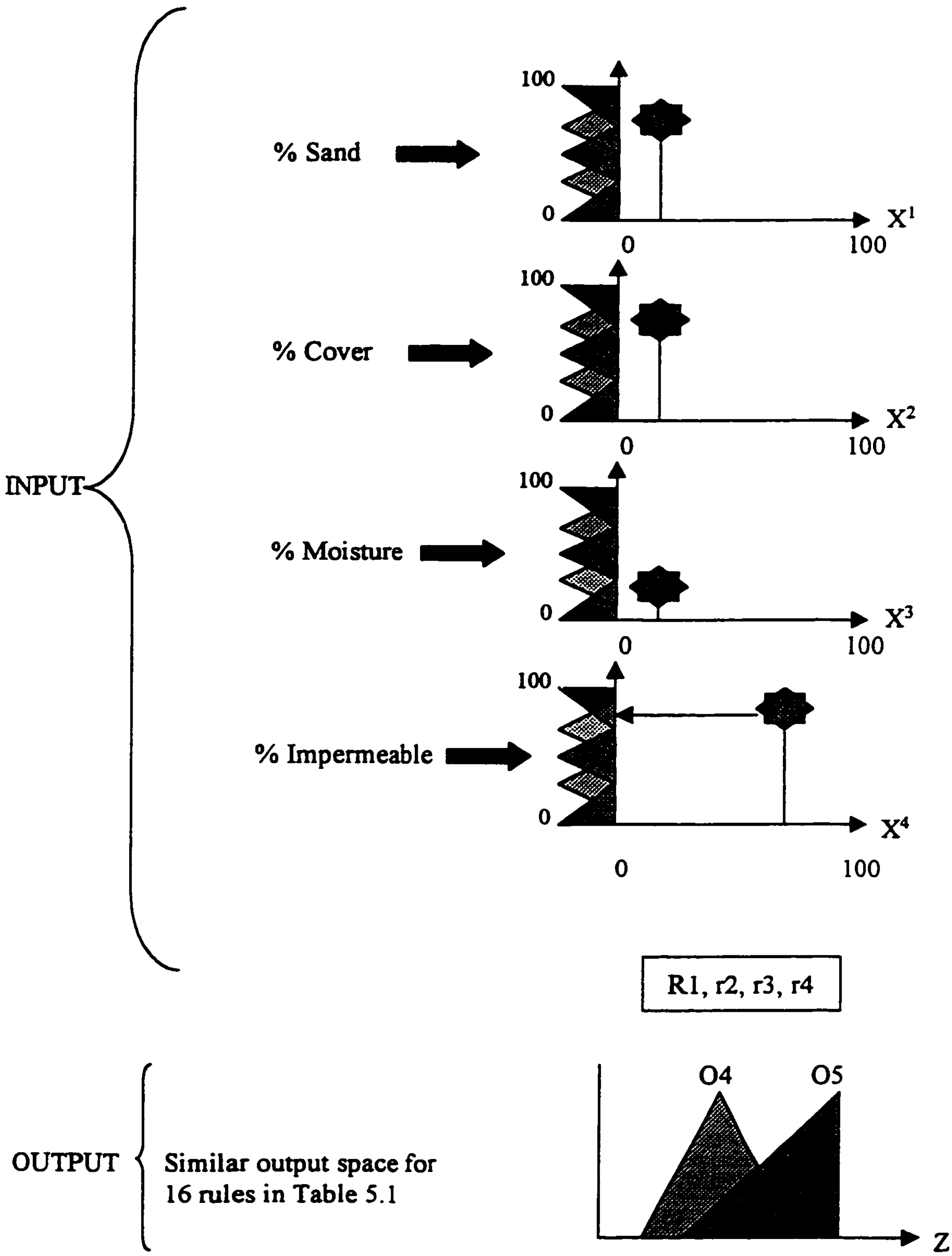


Fig 5.4 Effect of individual relations (r) in input space ($X^M{}_N$) on output space (Z)

Deductive reasoning is most useful when information must be obtained from the user in a context-dependent fashion, and amounts to an in-depth search of a decision tree. Inductive reasoning is usually useful when most of data are already present, or can be gathered in a context-independent fashion, and can be processed in batches in which fuzzy sets are used to represent ambiguities.

In serial rule firing the deductive method used in programming involves firing one rule at a time, and evaluating the effect of the rule firing after each step.

In parallel rule firing, and specifically in the inductive method parallel mode, all newly fireable rules are fired concurrently and no backtracking stack is maintained, since there are no fireable rules left unfired.

The reasoning strategy in FLEW^{CN} is implemented by parallel rule firing. This strategy is particularly useful when multi-variable fuzzy sets are employed. The general rule-firing scheme for one state rule-firing step in parallel form for FLEW^{CN} is shown in Fig 5.5. All fireable rules are fired at once, based on the user's interface strategy. Unfired rules are left over for backtracking.

The sequence of operations in a round of parallel rule firing is:

(i) A list is made of rules made fireable by the data and user specified control,

(ii) These rules are fired, and a list prepared of data modifications which call for the next action, and

(iii) Only those rules that relate to the specific space of inference are permitted to act and the rest are omitted.

A rule is fireable if the data yield an antecedent confidence above the rule-firing threshold. However, to actually be fired, the rule must also be turned on for firing (default condition), the rule must be picked for firing, and the program must be in run mode.

The number of rules which fire are proportional to the number of inputs. For example, assume that a single input and a single output are considered. During the rule evaluation for each action two rules should fire simultaneously to infer output action. By increasing the number of variables the number of rules increase exponentially (e.g., for a DISO model four rules should be fired at a time). Fig 5.5 shows five space for processes. The first and second

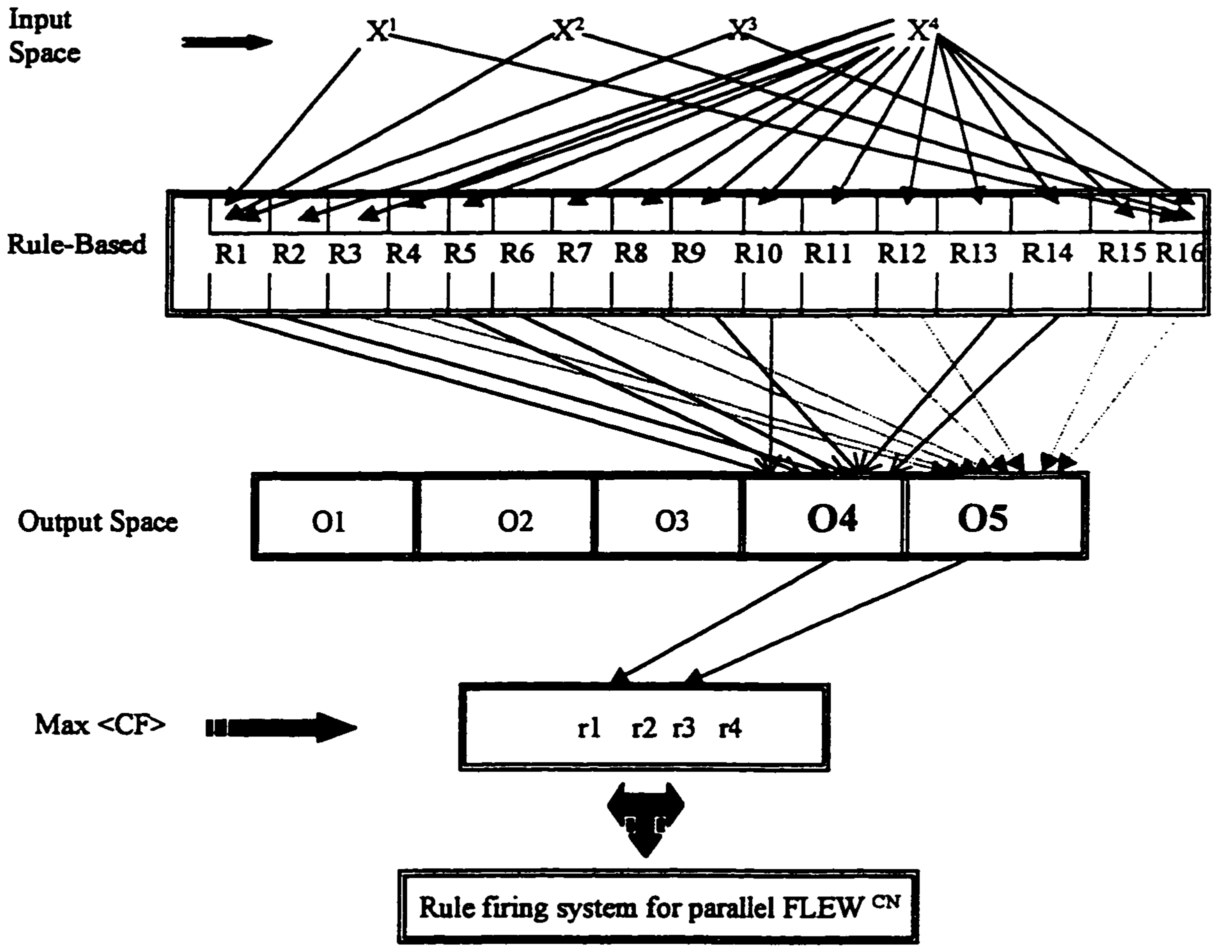


Fig 5.5 Rule-firing system based on parallel rule firing for MISO fuzzy logic model

phase of processes are presented in Fig 5.2, and Table 5.1, respectively. This example of a rule-based system has 16 rules. However, only two membership functions (O4 and O5] of output space [Z] have a maximum degree of acceptance to be the target in rule firing. In other words, other membership functions [O1, O2 and O3] have not been included in rule firing. When rules and output space are known then the rule with maximum <CF> is permitted to act in the defuzzification process. At the end of the process the output space will comprise Rf = [r1, r2, r4]. Therefore, Table 5.1 can be summarized into only four rules rather than sixteen rules, as given in Table 5.2.

The interpretation of Figs. 5.3, 5.4, and 5.5 are summarized in Fig 5.6. as an illustration of the whole process.

Table 5.2 Number of rules after reduction using a similar rules consequent.

R1:	**IF $X^1 = X^1_1 = g_{11}$ AND**		
	$X^2 = X^2_1 = g_{22}$ AND		
	$X^3 = X^3_1 = g_{31}$ AND		
	$X^4 = X^4_4 = g_{41}$ THEN	**→**	**Z = O4 ALSO**
R2:	**IF $X^1 = X^1_1 = g_{11}$ AND**		
	$X^2 = X^2_1 = g_{22}$ AND		
	$X^3 = X^3_2 = g_{31}$ AND		
	$X^4 = X^4_4 = g_{41}$ THEN	**→**	**Z = O4 ALSO**
R3:	**IF $X^1 = X^1_2 = g_{11}$ AND**		
	$X^2 = X^2_1 = g_{22}$ AND		
	$X^3 = X^3_2 = g_{31}$ AND		
	$X^4 = X^4_5 = g_{41}$ THEN	**→**	**Z = O5 ALSO**
R4:	**IF $X^1 = X^1_2 = g_{11}$ AND**		
	$X^2 = X^2_2 = g_{21}$ AND		
	$X^3 = X^3_2 = g_{31}$ AND		
	$X^4 = X^4_5 = g_{41}$ THEN	**→**	**Z = O5.**

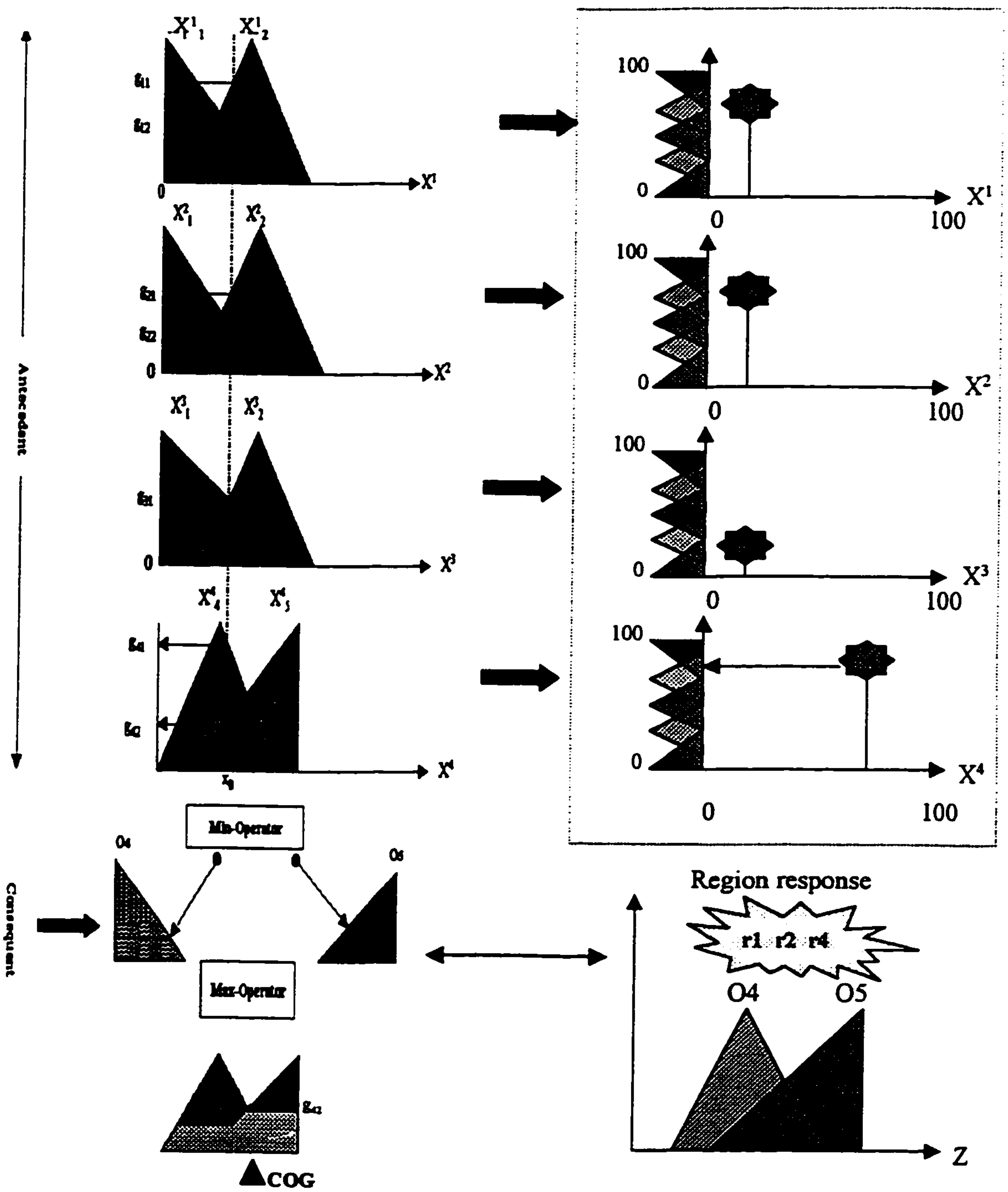


Fig 5.6 The processes of rule reduction due to similarity properties of the response region.

5.6 User Interface

Monitor definition for an application system of interest is a two-step process, which comprises construction of a model followed by definition of the user interface. It is well known that programing of a user interface is difficult. Myers (1992) stated that studies consistently show that the user interface portion comprises about 50% of the code and development time. The most important characteristics of a user interface is simplicity, visibility of the course of action, transparency of the system, and pre-visibility of the effects of user action for flow of I/O. In large scale software development it is recommended that the user interface should be independent of the program. Some programs can be connected to the interface with text I/O (non-graphical) in the form of a question and answer strategy. The process of I/O has to be done in the screen rather than on the keyboard. This is more useful in a conventional expert system. However, in a fuzzy logic algorithm there is a different style of visualization of flow of input and output. The most popular one is the presentation of I/O as an event. When the user hits a keyboard or a mouse button, an event, or reaction, appears in a task window. This kind of user interface is very simple but is also dependent on the main program. In other words, any changes in the main program may necessitate changes to the user interface.

In computer programming terms, each parallel rule firing is highly non-procedural in that the sequence of rules has nothing to do with what happens and the response is determined solely by the data. This is difficult to program in most computer languages as almost all common computer languages (FORTRAN, COBOL, BASIC, C++) are procedural; only a few AI languages such as PROLOG, LISP and FLOPS are non-procedural.

The FLEW^{CN} user interface model contains the graphics routines used to display and simulate the I/O of the model graphically. To display it in the graphics mode it uses the EVAGVA.BGI (Borland 1994) library.

This user interface is designed for ease of user operation to operate simply and is based on keyboard commands with C language capability. This program is externally switched to algorithm A, B or C (previously mentioned). The structure of the routines is as follows:

(i) Definition of the function that is used for echoless keyboard entry which includes the

CURSOR and FUNCTION keys (UP, DOWN, HOME, END, LEFT, RIGHT, PGUP, PGDN) on the keyboard.

(ii) "Initialize Graphics" is used to initialize the graphics library. An initialize graphics system test results for visualizing the output on the screen.

(iii) "Draw Graph" is used to initialize a graphical plot area on the VGA monitor screen and save the current viewport setting boundary around the bar controller.

(iv) Display text mode in the screen such as title, variables, membership function, etc. on the screen.

(v) "Initialize Coordination" is used to draw the axes and coordinates of each variable used in the inputs and output.

(vi) "Plot Graph" is used to plot a fuzzy set $X_N[]$ in the graph plot area on the screen.

(vii) Transform the universe $X^M[\ ]$ and sets $X_N[\ ]$ into a normalized universe $x = [0,..,255]$ having a fuzzy set [U_SET] with transfer function $X_w = 2.55$.

(viii) Draw and plot a UoD [U_SET].

(ix) "Plot Singleton" is used to plot a singleton value onto the fuzzy set Xo[] which was previously plotted in the graph plot area on the screen. The "Plot Singleton" routine uses a set write mode (XOR_PUT) to draw graphics to the screen.

(x) Calculate the offset for a singleton based on the scale of output and draw a singleton for all event inputs and resulting output.

(xi) Indicator for showing the change of input variables that change the output based on fuzzy logic open loop control.

(xii) " Initialize Bar" is used to initialize a bar plot area and the borders on the VGA screen to predefine inputs and output.

(xiii) "Draw Bar" is used to plot a fuzzy set $X_N[\ ]$ in the bar plot area as pre-defined under (xii) above on the screen.

(xiv) "Initialize Controller" is used to simulate the startup routines that would be performed if this was an actual controller. In this case, it simply initializes the VGA display and sets up the user interface.

(xv) Set color for controller text display.

(xvi) "Get Input" is used to simulate the reading of the input variables for the fuzzy logic controller.

(xvii) "Set Input" is used to simulate the setting of the output variables for the fuzzy logic controller. In this case, however, it updates the user display and retrieves new input.

(xviii) Plot new action results from inference engine program output.

5.6.1 Visualization of I/O

The style of graphical presentation of input and output on the screen depends on the quantity and quality of data. Although large quantities of information become available on-line, information itself is useless without effective access mechanisms. One form of information can be displayed using information murals. This kind of presentation not only shows relative position and size of the focus area, but also gives the user an attractive view of the whole of the system on the screen. Color, in this method, is used to highlight attributes in line and bar forms. As indicated in the previous section, Eqs 5.1 to 5.9 are presented as line display membership functions in the I/O space. The control action of the input bars represent the rate of change of inputs that determine the output result. The example presented in Sec .5.4, in Figs 5.2 to 5.6 and in Tables 5.1 and 5.2 is presented in visual form in Fig 5.7 (main screen of the user interface). In this display inputs are sand (25%), cover (19%), moisture content (18%) and impermeable area (78%). As it appears on the screen, the output bar shows that CN = 92. This output is calculated based on the methodology presented in Ch.3. In the second display (Fig 5.8) input is sand = 50% (partially LS and partially MS), cover = 50% (partially LC and partially MC), moisture content = 80% (partially FC and partially Saturation) with the presence of 59% impermeable area. The curve number shown appears as partially HCN and partially VHCN. However, the crisp response to such inputs is a value of 85. This value represent the result of defuzzificd output space as a curve number. Validation of the estimated curve number is discussed in the next section.

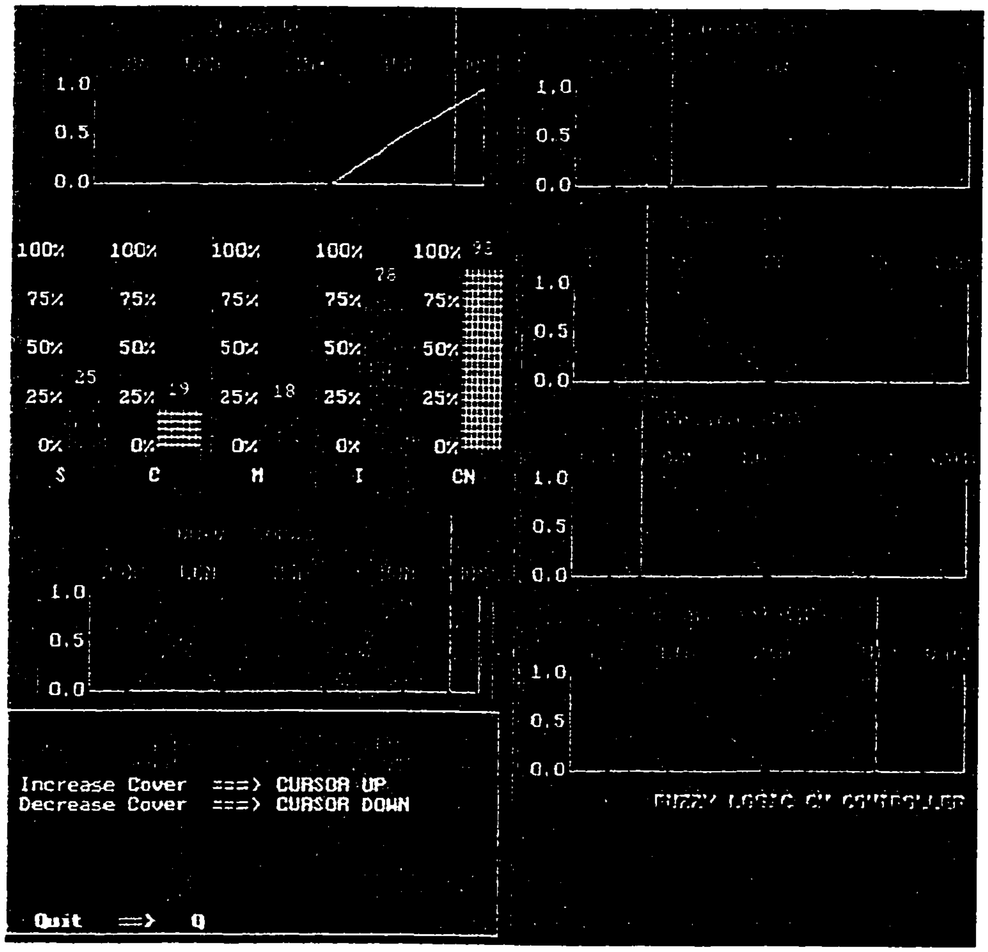


Fig 5.7 Visual display of user interface for inputs and output given in Table 5.1.

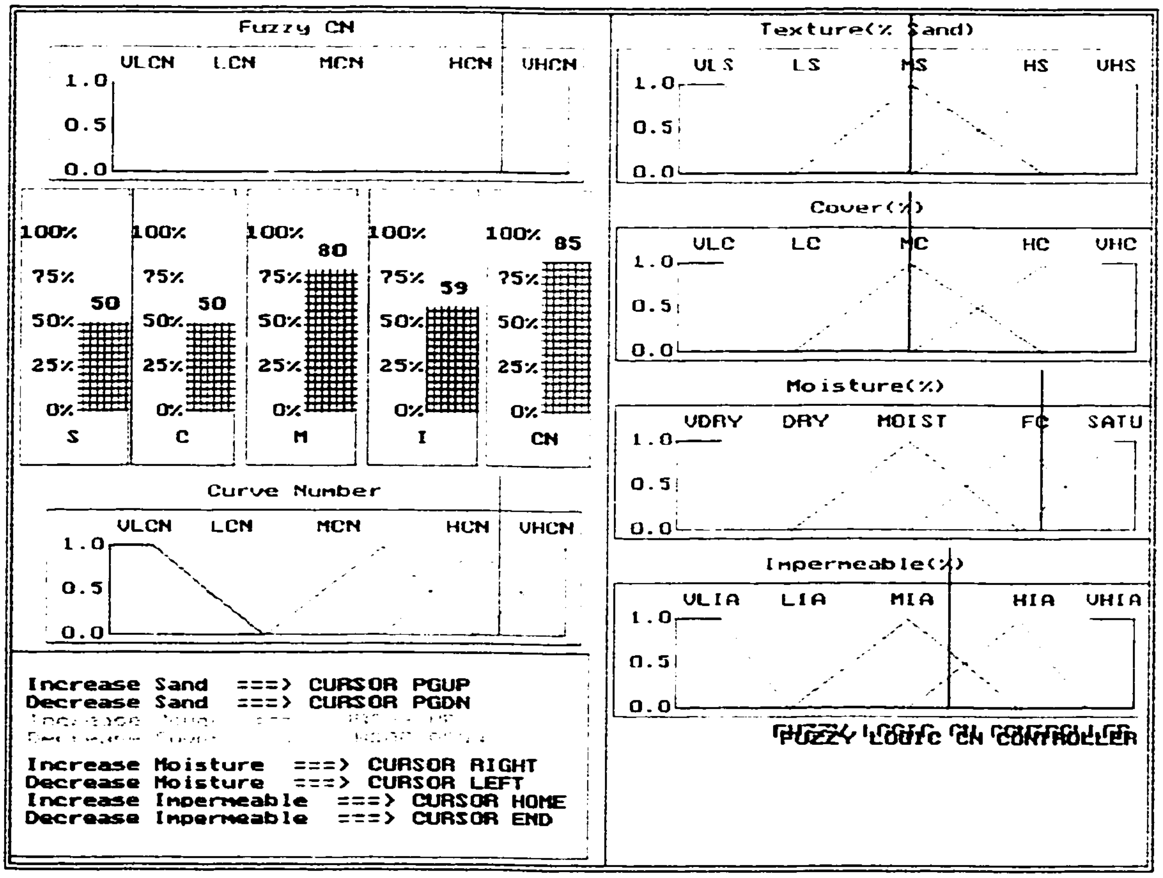


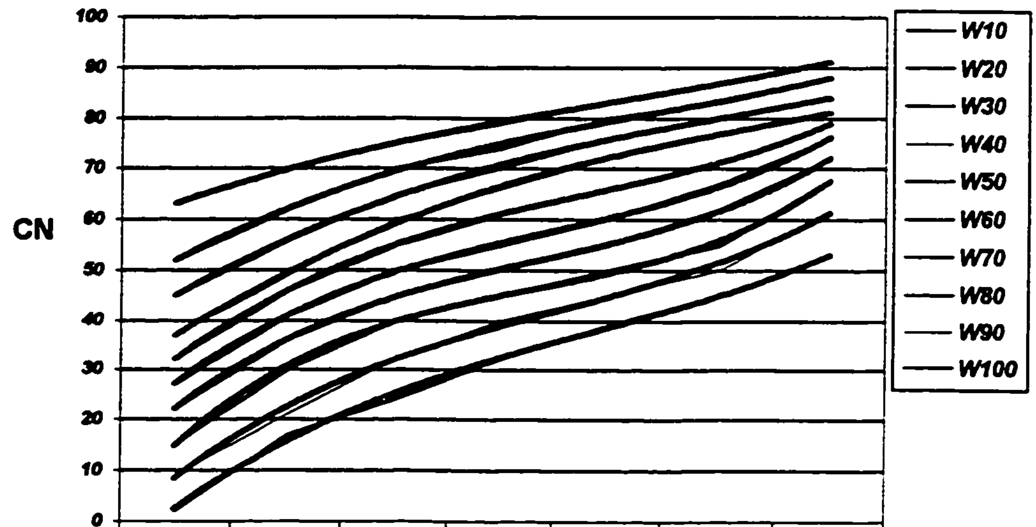
Fig 5.8 Visual display of user interface for a set of inputs differing from those of Fig 5.7.

5.7 Quantification of CN for Forest and Rangeland

A particularly onerous task is the estimation of the curve number based on land use. In practice, unlimited variations in land use exist in terms of management and cover. In the SCS method specific land uses are considered such as fallow (an agricultural land use with highest runoff potential), row crop, small grain, close-seeded legumes, grassland, meadow, woodlands, and forest. Specifically the SCS method defines woodland as an isolated grove of trees being raised for woodlot use and visually evaluated in terms of hydrologic condition. SCS classifies woodlands in one of three fuzzy classes as Poor, Fair or Good condition, as described in Sec.2.2.

As an example of the application of the FLC method, assume a woodlot with ten classes of cover condition, from W10 (10% wooded cover) to W90 (90% wooded cover) and with twelve soil classes, as represented by a rectangular soil classification. Soil ranges from sandy soil (more than 80% sand) to clay soil (less then 30 % sand or having more then 60 % clay). By applying the same methodology as presented in Ch.3, the result of the simulation for a wooded area is presented in Fig 5.9 As shown in Fig 5.9, for sandy soil with wooded cover, the curve number ranges from CN=2 (W90) to CN = 62 (W10). The highest CN values obtained are for low wooded cover (W10) and clay soil (60% clay) for which the CN values range from 62 to 91.

Another example is the generated curve number for a rangeland area. Again, the SCS method categorizes rangelands into three hydrological conditions of Poor, Fair or Good as for woodland areas in the previous example. A rangeland area can be evaluated in the form of a continuity of % cover density and soil texture (% sand). The results of simulation are presented in Fig 5.10. As may be seen, the response of the rangeland and woodland is similar when based on % cover density and soil texture. Figs 5.11 and 5.12 show a comparison between generated curve number (CN) for rangeland and woodland for sandy soil and clay soil (clay $\leq$ 60) respectively. Cover density and soil are the same, only the types of cover are different. Quantitatively, it can be concluded that the percentage of cover density is more important then the type of cover.



	Sand	SL/LS	L	SIL	SCL\SICL	SC\SIC	Clay
W10	63	70	75	79	83	87	91
W20	52	62	70	74	80	83	88
W30	45	56	65	70	76	80	84
W40	37	49	60	66	72	77	81
W50	32	46	55	61	66	71	79
W60	27	41	50	55	60	67	76
W70	22	36	45	50	55	62	72
W80	15	30	40	45	50	55	68
W90	9	21	32	40	45	50	62
W100	2	17	24	32	39	45	53

Fig 5.9 Relationship between soil texture, woodland area (ten classes from 10% to 100%) and curve number based on rectangular soil classification method

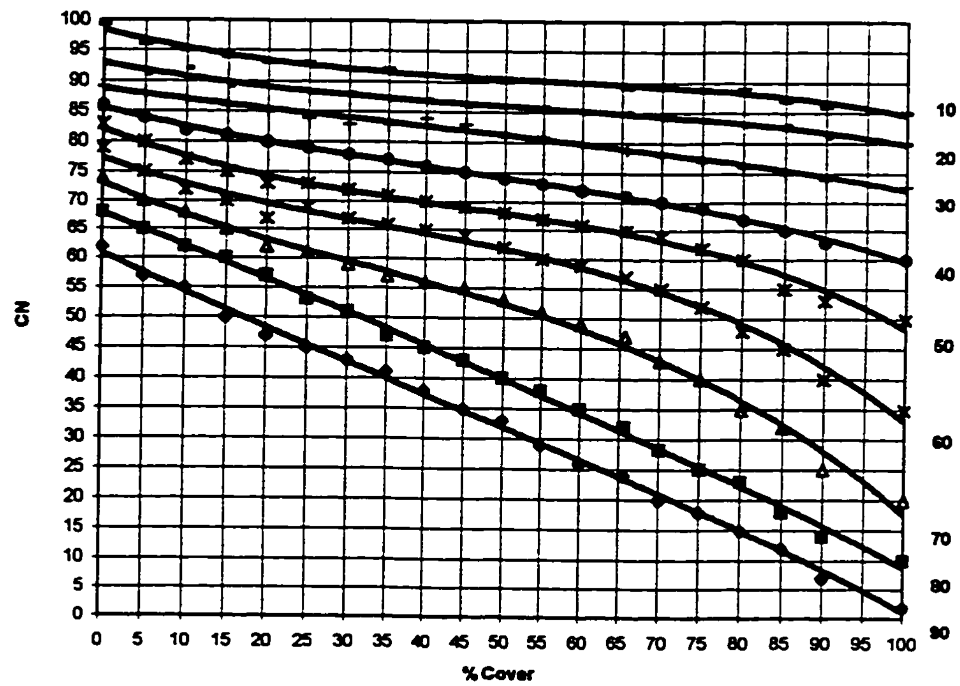


Fig 5.10 Generated curve number for rangland based on soil texture, and cover density .

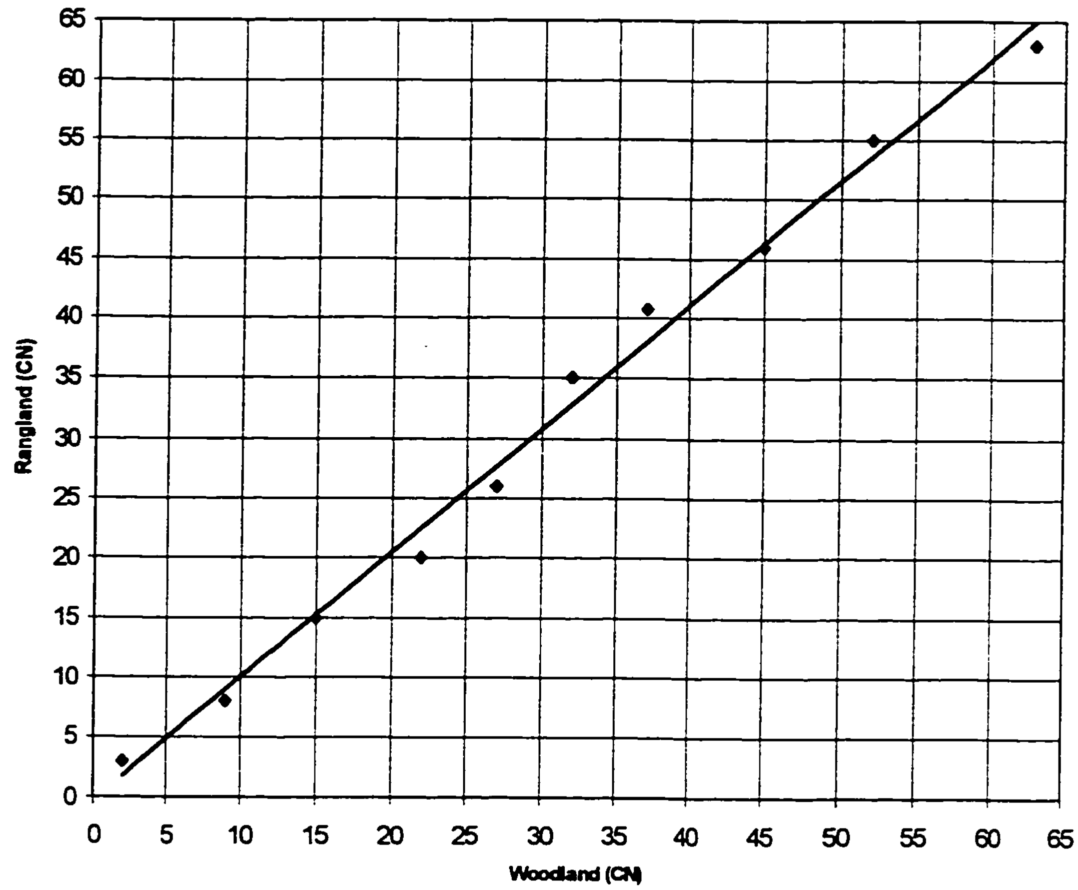


Fig 5.11 Comparison between curve number generated for woodland and rangeland area for sandy soil.

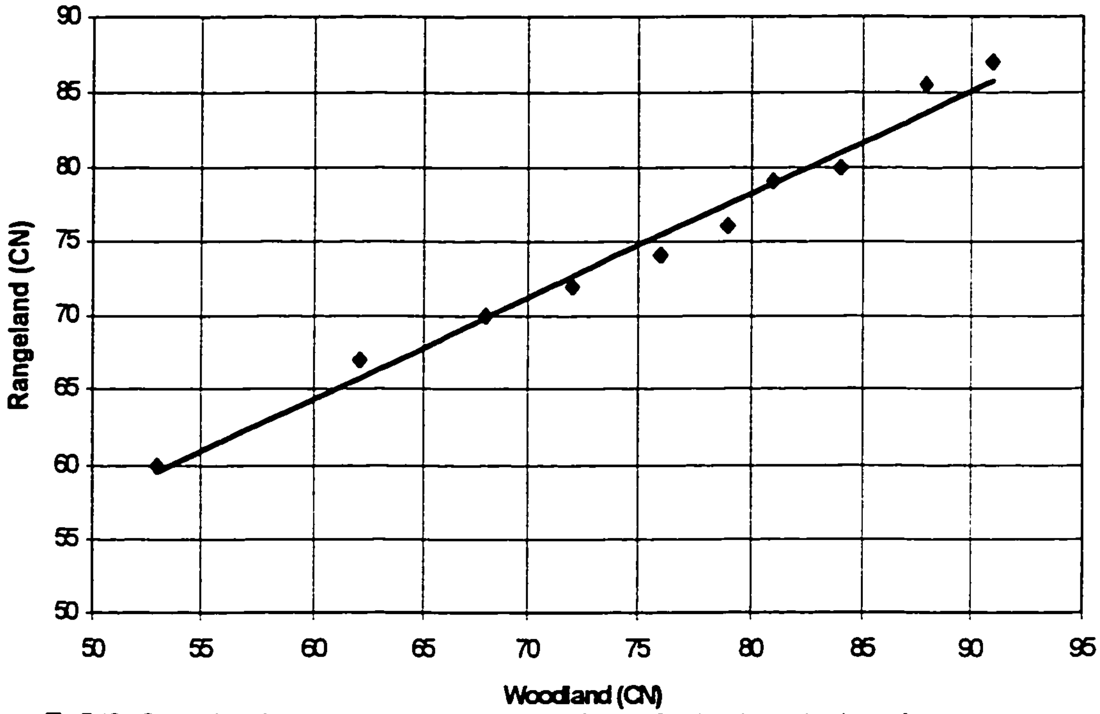


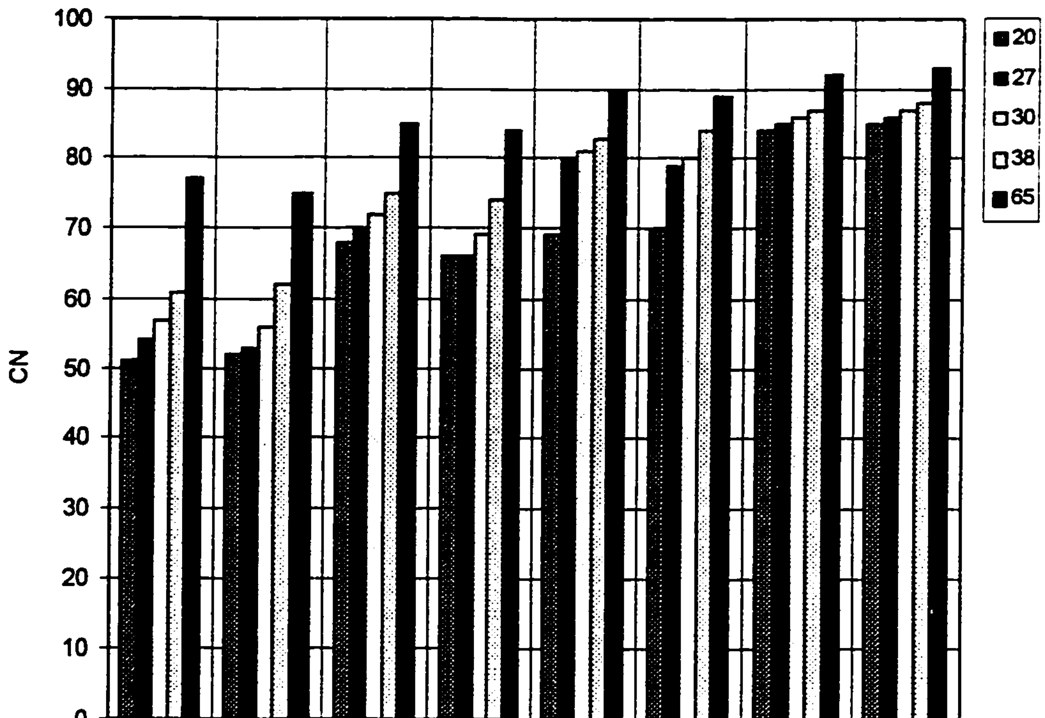
Fig 5.12 Comparison between curve number generated for woodland and rangeland area for clay soil (clay = 60%).

5.8 Validation

Development of an environmental model, whether static or dynamic, is an activity that generally involves two phases: first by creation of a model or modification of an existing model, and second by comparison of the model with the real system modeled. This second phase is known as model validation. In conventional modeling cross-validation between observed and estimated values (statistical modeling) is a common way to validate a model (Cressie 1993). In expert systems the main guiding principle might be called '*expert knows best*'. However, this approach is often infeasible; an example being that of validation of a logical model. For this reason one of the most difficult and controversial tasks in artificial intelligence development is validation and verification of an expert system. Lee and O'Keefe (1994) argued that there are numerous possible methods for expert system verification and validation. However, an artificial intelligence product only can be verified based on explicit objectives and final conclusions. Lehner (1989) further speculated about expert systems for which there are no human experts or experience to evaluate and to verify the response.

In view of the above, for evaluation of the response of the FLC model a comparison was made with the SCS model presented as a benchmark for this study. Again it is noted that the SCS method is a discrete model, whereas the FLC model is a continuous model.

A comparison was made between the SCS method and the FLC method for calculating the curve number for an urban area, based on % impermeable area and soil texture. Figs 4.14 and 4.15 present results of a 10 x 10 FAM for % impermeable area and soil texture (% sand) with equal membership function domains with 50% overlap. The SCS method was used to simulate five different percentage impermeable area classes for urban areas (20%, 27%, 30%, 38%, and 65%) in four discrete hydrologic soil groups A, B, C, D. The results of the SCS method application were compared with Fig 4.15 which was generated using the FLC continuous model for hydrologic soil groups A, B, C, and D. The results of this comparison are presented in Fig 5.13. As may be seen, the response of the FLC method to % impermeable area is close to that of the SCS method for 65% impervious area and hydrologic soil group D, but differs most for hydrologic soil group B. It was expected that the results would be different because of the nature of the model predications. The FLC model is based



	FLC	SCS	FLC	SCS	FLC	SCS	FLC	SCS
20	51	52	68	66	69	70	84	85
27	54	53	70	66	80	79	85	86
30	57	56	72	69	81	80	86	87
38	61	62	75	74	83	84	87	88
65	77	75	85	84	90	89	92	93
	A		B		C		D	

Hydrologic soil groups

Fig 5.13 Comparison between SCS observed data and estimated data from FLC for hydrologic soil groups and impermeable area.
Note: A, B, C and D are hydrologic soil groups.
SCS: Presented values by SCS method
FLC: Estimated values by FLC model

on fuzzy logic as a continuous translator of fuzzy data, whereas the SCS method is a discrete model based on empirical small watershed data. The FLC model has no limitation to any combination of % impermeable area and soil texture for predicting curve number.

6. DISCUSSION

The program FLEW^{CN} was developed based on logical relationships between watershed parameters rather than on experimental data. The general form of the development was an integration based on of three areas of knowledge viz. hydrology, expert system, and fuzzy logic. The model was developed in three phase of: SISO model (individual relationship of % sand and % clay with curve number), DISO models (soil texture and each of the other parameters in turn with curve number output in three dimensional space), and finally a MISO model called Fuzzy Logic Expert Watershed Curve Number (FLEW^{CN}). FLEW^{CN} is based on a DOS platform and comprises a mixed algorithm composed of the conditional "*IF.. AND... THEN*" clauses and unconditional "*IF.. AND... ALSO...THEN..ONLY*" clauses. The latter give specific power to specific variables.

The strategy in development of a traditional expert system is to use one of the more sophisticated artificial intelligence languages such as PROLOG (a declarative language) or LISP (a list processing language). However, fuzzy logic algorithms are more easily adapted to integration in a procedural language (algebraic language) such as C or Pascal, etc. The program FLEW^{CN} is coded in the Turbo C language by combining three algorithms in sequential procedural processes of initialization, processing and graphical representation of inputs and output (user interface). The program is in the form of a fuzzy control procedure rather than that of a conventional program and is user-friendly in application, even for a person with no knowledge of fuzzy logic. From a programming point of view, the fuzzy logic algorithm is easy to understand; however it was complex to program and to visually show an interaction of input and output space. Capture of the knowledge was a complex process, in particular in dealing with the interactions of the variables (e.g., soil texture has an effect on cover, moisture content, infiltration, etc.). From a programing point of view this would be an impossible task for any but professional computer programmers without tools such as an expert system shell. However, programs developed using an expert shell depend on the specific expert shell for execution. At present, no expert shell exists which manipulate fuzzy data in a manner similar to that of conventional expert systems. However, some fuzzy expert shells can generate C code based on the specific user defined model; however, this type of approach does not allow for interaction between fuzzy data and a user interface. The advantage of $FLEW^{CN}$ is that it is independent of any expert shell or any other program. It also provides some of the structures that can increase efficiency in use of the model, such as visual demonstration of inputs and output.

The program is organized as a procedural process with elements of action (membership functions). Each membership function is defined in binary format as an object of an element. Each element has two different references of pivot (only a specific element appears to act when the program calls for action) and global (all of the elements are affected when the program calls the elements). This process appears during execution time as a user interface.

The user interface provides a fuzzy control program to answer questions of the "How much?" type (quantity) by key action to show the quantity of input's space and inference action in the output space. This feature provides an effective means to assess the effect of changes in combination of inputs to reach an output response.

An unconditional assertion is used for fuzzy set "saturation" (membership function) and fuzzy set "very high impermeability". The certainty factor <FC> is characterized by key words which represent the importance of the degree of the rules. For example, when several rules in the data-base are designated for action during rule firing, a particular rule has a maximum effect to associate with the response space when it has a linguistic definition "always" or CF=1. If a membership function has a minimum effect on the consequent the associated FC is a minimum, of CF=0. (never). These processes occur as inference processes during rule firing.

In a conventional expert system the user deals with such questions as "what?", "what if?", " how much? " and "why?". This program is designed only for quantitative (how much) inputs and output. If a question does not fit within this strict boundary of limitation, then the program does not have the ability to provide an answer.

One of the most powerful, untapped, capabilities of computational intelligence technology is in dealing with solutions to complex phenomena, by applying adaptation and estimation based on the experience and logical relations of parameters. In this project, computation of logical relationships between parameters was built into the knowledge-base as an integral part of the way in which the system works.

For some specific limited intervals the curve number computed by the SCS model and $FLEW^{CN}$ were close. However, the $FLEW^{CN}$ program is not limited to the prediction of curve numbers for only specific intervals, but is able to predict curve number for all combinations of inputs. $FLEW^{CN}$ comprises four inputs and one output. The universe of discourse for each input ranges from 0 to 100. This means that $FLEW^{CN}$ has a knowledge-base of $(100)^4$ inputs which is used to predict the output response.

This program differs from other expert systems that have been developed in the field of hydrology. In general, an expert system is commonly developed based on the experience of a human expert, or it can be merged with another conventional program. $FLEW^{CN}$ is a stand-alone program which uses fuzzy control methodology that is more conventionally used in industrial applications.

This program has incorporated in it an innovative addition to conventional fuzzy expert system technology in using inputs based on parallel rule firing and having a parallel structure. Each variable with its membership function is first compared in the output space in parallel with the other variables to reduce the number of rules, before the inference process continues with a max-* operator to reach a conclusion. This process of reasoning is able to be used in a fuzzy expert system by utilizing the ability to manage the inference engine.

7. CONCLUSIONS

In the introduction to this dissertation, needs were identified for finding tools that could contribute to the development of hydrologic design. Fuzzy set theory and fuzzy logic control were the be used to develop a program to estimate a watershed curve number. Advanced technology and artificial intelligence in the field of computer programming, and recent theory of uncertainty to quantify fuzzy data, are seen to hold great promise in pursuing the goal of improved hydrological modeling. An open-loop fuzzy control (FLEW^{CN}) model was developed from different intermediate fuzzy logic models.

There are a number of significant areas in which the development of this fuzzy logic expert system for watershed curve number computation contributes to urban and suburban hydrology and computational intelligence. These are:

(i) The algorithms (A, B and C) developed for multivariable fuzzy control, decomposed a complex fuzzy controller into several simple fuzzy controllers. This approach results in a smaller number of control rules than the traditional multivariable fuzzy controller while maintaining the power of parallel computing and rule firing. Furthermore, the new approach resulted in a complexity no in not greater than that of the simple fuzzy controllers used.

(ii) the use of fuzzy data (quality data) pertaining to the conceptually well-known hydrologic indicator of watershed curve number was based on logical relationships rather than human expert experience. The methodology of using an open-loop fuzzy logic control is a new idea in hydrology and environmental science. This research builds a bridge between computational intelligence and watershed hydrology.

(iii) In the field of computational intelligence the capturing of knowledge, organization of knowledge, processing of data, and visualization of a response in a simple way is critical. The development of FLEW^{CN} has shown the ability to present all of the relevant terms in only one task window.

(iv) The generality of the development indicates that the methodology used is not only applicable to curve number estimation, but also to other technological applications. It may, for example, be used to control other fuzzy data sets to reach a crisp response, as in the design of irrigation systems, ventilation, etc.

(v) Validation and verification are difficult tasks in an artificial intelligence domain. The $FLEW^{CN}$ program is a logicaly-based program and was validated by comparing its response to that of the SCS method which is a well-accepted empirical model. The comparison between $FLEW^{CN}$ and SCS validated the use of fuzzy logic control in the form of a continuous model versus the empirical and discreet SCS model.

(vi) $FLEW^{CN}$ provides a logical step forward in the development of tools for estimation of complex and time-consuming hydrologic parameters, and is particularly useful for urban areas with portion of impermeability. The curve number index is a logical way to estimate not only runoff potential but also for estimating peak flow.

(vii) The graphic on-screen visual response of $FLEW^{CN}$ to inputs enables enhanced communication between a user and the program. In this way the program provides for a greater understanding of the underlying theory .

(viii) Based on the comparison with the SCS method and the program performance, it is recommended that $FLEW^{CN}$ be applied to urban and suburban areas to obtain a logical prediction of the curve number.

(ix) Based on the predicted values for woodland and rangeland, it may be concluded that cover density is more important than type of cover for prediction of the curve number.

8. FUTURE RESEARCH

Given the complexity of the task of study (multi-relational variables with multi-effects), $FLEW^{CN}$ does not include an evaluation of the curve number for all possible land uses. This is because the program has a data base limited by time constrains. There are significant areas where further development is required through use of multi-task windows with different land uses. These land uses could be integrated as specific expert systems to estimate not only curve number (CN) but also infiltration and other desired parameters in the form of a MIMO. $FLEW^{CN}$ is not an empirical nor a physical model but is based on logical relations between the watershed parameters. This advantage gives such a model a more practical and dynamic prediction ability for parameters that are changed by time such as moisture content and cover density. With further development the system will allow the user to not only estimate hydrologic parameters but also to obtain expert advice. Another area in which this algorithm could be used is that of automatic control of sprinkler and drip irrigation systems.

9. REFERENCES

Adams, W. H., G. G. Siekinen, and K. Holland. 1992. Development of the water Efficient Landscape Planner Knowledge-Base. Proceedings of 4th international Conference. ASAE Computer in agricultural Extension Programs. Orlando, Florida. pp. 762-767.

Albert J., F. Ferri, Domingo J., and M. Vicens. 1990. An approach to natural scene segmentation by means of genetic algorithms with fuzzy data. In Pérez de la Blanca N., SanFeliu A., and Vidal E. (eds) Fourth National Symposium in Pattern Recognition and Image Analysis (Selected Papers), pp. 97-112. World Scientific Publishers Co. Inc., Singapore.

Ameskamp, M. 1997. Three-Dimension Rule-based Continuous Soil Modelling. Unpublished Ph.D. thesis, Iinstitut für informatik und praktische mathematik Christian-Albrechts-Universität, Kiel, Germany.

Arbour, J. H. 1992. Development and Validation of an Expert System for Soil Erosion Control Planning in Prince Edward Island, Unpublished Ph.D. thesis, Department of Agricultural Engineering Technical University of Nova Scotia, Halifax. Canada

Arnold, J. G., J. R. Williams, A. D. Nicks, and N. B. Sammons, 1990. SWRRB: A Basin Scale Simulation Model for Soil and Water Resource Management. 1st ed., Texas A&M University Press. Collage Station, TX.

Arnold, J. G., J. R. Williams, R. H. Griggs, and N. B. Sammons., 1991. SWRRBWQ- A Basin Scale Model for Assessing Management Impacts on Water Quality. USA, A governmental Research service, Grassland, Soil and Water Research Laboratory, Temple, TX.

Arnold, J. G., J. R. Williams, R. Srinvansan, K. W. King. And R. H. Griggs. 1995. Soil and Water Assessment Tool (SWAT). USDA, Agricultural Research Service, Grassland, Soil and Water Research Laboratory, Temple, TX.

Aron, G., A. C. Miller.,and D. F. Lakatos. 1977. Infiltration Formula Based on SCS Curve Number. ASCE, Journal of the Irrigation and Drainage division. Vol. 103, No. IR4, pp. 419-427.

Bardossy A., I. Bogardi, and L. Duckstein. 1990. Fuzzy regression in hydrology. Water Resources Research, Vol. 26, No 7, pp. 1497-1508.

Bardossy A. and M. Disse. 1993. Fuzzy Rule-based Models for Infiltration, Water Resources Research, Vol. 29, No 2, pp. 373-382.

Bardossy. A., A. Bronstert and B. Merz. 1995. 1-, 2- and 3-dimensional modeling of water movement in the unsaturated soil matrix using a fuzzy approach, In Advances in Water Resources, Vol. 18. No.4, pp. 237-251.

Bardossy, A. and L. Duckstein. 1995. Fuzzy Rule-Based Modeling with Applications to Geophysical, Biological and Engineering Systems CRC Press, Boca Raton.

Barr, V. Pratt, I. 1988. Exploring expert systems. New Production Reviews. J, Computer pp. 68-78.

Beasley, D. B., L. F. Huggins, and E. J. Monke, 1980. ANSWERS Areal Nonpoint Source Watershed Environment Response Simulation: A model for watershed planning. Transactions, ASAE 23(4) pp. 938-944.

Beasley, D. B., and L. F. Huggins, 1991. ANSWERS Users Manual 2nd ed. Agricultural Engineering Department Publication No. 5.

Berenji, H. 1992. A Reinforcement Learning-Based Architecture for Fuzzy logic control. Int. J. Approximate Reasoning. Vol. 6, No. 2, P: 267-292.

Bezdek J. C. and R. J. Hathaway. 1994. Optimization of fuzzy clustering criteria using genetic algorithms. In Proc. First IEEE Conference on Evolutionary Computation (EC-IEEE'94), pp. 589-594. Orlando, FL.

Bonarini A. 1993 . Learning incomplete fuzzy rule sets for an autonomous robot. In Proc. First European Congress on Fuzzy and Intelligent Technologies (EUFIT'93), pp. 69-75. Aachen.

Borland International 1994. Turbo C++ user guide, Inc, Scotts Valley, CA.

Boughton, W. C. 1989. A review of the USDA-SCS Curve Number Method. Aust. J. Soil Res. 27, pp. 511-523.

Buckley J. J. and Y. Hayashi. 1993. Fuzzy genetic algorithms for optimization. In Proc. International Joint Conference on Neural Networks (IJCNN'93), pp. 725-728. Nagoya, Japan.

Bullock, P. R., E. Dejong., and J. J. Kiss 1990. An assessment of rainfall erosion potential in southern Saskatchewan from daily rainfall records. Can. Agric. Eng. Vol, 32. No, 1. pp. 17-24

Burney, J. R., 1977. Manual for small catchment distribution parameter hydrology model version 3. Unpublished report, Dept. of Agric. Eng. Purdue University, West Lafayette, IN.

Burney, J. R., and L. M. Edwards, 1992. Monitoring and modelling soil erosion/conservation in the upper Wilmot river watershed, Prince Edward Island. COSSEM (COol Season Soil Erosion Model) V 1.0. Structure and example applications. Dept. of Agric. Eng., Technical University of Nova Scotia, Halifax, Nova Scotia, Canada.

Burrough, P. A. 1989. Fuzzy mathematical methods for soil survey and land evaluation, J. of Soil Sci., 40, pp. 477-492.

Burrough, P. A., R. A. MacMillian. and W. van Deusen. 1992. Fuzzy classification methods for determining land suitability from soil profile observations and topography. J. Soil Sci. 43. pp. 193-210.

Castro, J. L. 1995. Fuzzy Logic Controllers are Universal Approximators, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 25, No. 4, pp. 629-635.

Chen, C. L. 1982. Infiltration Formulas by Curve Number Procedure. Journal of the Hydraulics Division, ASCE, Vol. 108. No. HY7. Pp. 823-829.

Christiansen, E. 1997. Watershed Modelling System (WMS), Engineering Computer Graphics Laboratory at Brigham Young University, UT.

Cook, D. J., W. T. Dickinson, and R. P. Rudra. 1985. Games: the guelph model for evaluating effects of agricultural management systems on erosion and sedimentation. User's manual. Technical Report 126-71. p.75 University of Guelph, Ontario.

Cox. E. 1992. Fuzzy Fundamentals. IEEE Spectrum, pp. 58-61.

Cox E. 1993. A model-free trainable fuzzy system for the analysis of financial time series data with fuzzy set morphology and rule association optimization through a genetic optimizer. In Proc. Second Annual Conference on Artificial Intelligence Applications on Wall Street: Tactical and Strategic Computing Technologies, pp. 280-285. New York.

Cressie, N. 1993. Geostatistics: A tool for Environmental Modelers. Editor : Goodchild, M. Ch.41, pp. 414-421.

DaRe. A. E. E., 1993. An Expert System for Groundwater Supply Protection and Management. Unpublished M.A.Sc thesis, Department of Civil Engineering, Technical University of Nova Scotia, Halifax, Canada.

David, R. 1989. "Analysis of Forest Succession with Fuzzy Graph Theory", Ecological Modeling, 45, pp. 261-274.

Davidson, V. J 1997. Fuzzy control for food processing. Smittal, G. S. (Ed) Computrized Control Systems in the Food Industry. Ch.6, pp. 179-234. Marcel Dekker, Inc. New York, NY.

Decoursey, F., and M. W. Snyder. 1969. Computer-oriented method of optimizing hydrologic model parameters. Journal of Hydrology. 9, pp. 34-56.

Dermo, A. P. J, C. G. Wesseling, N. H. D. T Cremers, R. J. E. Offermans, C. J. Ristsema and K. Van Oostindie 1994. LITEM: a new physically-based hydrological and soil erosion model in a GIS-environment, theory and implementation. Variability in stream Erosion and sediment Transport Processing of the Canberra Symposium. IAHS Publ. No 224. pp. 439-448.

Duckstein, L., Bogardi, I. and A. Bardossy, 1990. Tutorial: Fuzzy Set Theory in Water Resources Systems, in Optimizing the Resources for Water Management, Edited by R.M. Khanpilvardi and T.C. Gooch, pp. 434-441, ASCE.

Elliot, W. J., A.M. Liebenow, J. M. Laflen, and K.D. Kohl, 1989. A compendium of soil erodibility data from WEPP cropland soil field erodibility experiments 1987 & 88. NSERL Report No. 3. The Ohio State University and USDA Agricultural Research Service.

Embleton, K. M. and B. A. Engel 1992. An artificial intelligence based farmstead assessment program for ground water quality concerns. Proceedings of 4th international Conference. ASAE Computer in Agricultural Extension Programs. Orlando, Florida. Pp.750-761.

Engman, E. T., A. Rango, and J. Martince.1986. EXSRM : An expert system for Snowmelt Runoff Model (SRM) Proc. Baltimor Symposium, IASH Publ. No. 181, pp. 417-226.

Feigenbaum, E. A. 1977. The art of artificial intelligence: Themes and case studies of knowledge engineering. International Joint Conference on Artificial Intelligence. Vol. 5, pp. 1014-1029.

Ferreira, V. A. and R. E. Smith., 1988. The limited physical basis of physically based hydrologic models, In: Modelling Agricultural, Forest, and Rangelands Hydrology, ASAE, publication 07/88, ST, Joseph, MI.

Ferson, S. and R. Kuhn. 1992. Propagating uncertainty in ecological risk analysis using interval and fuzzy arithmetic. Computer Techniques in Environmental Studies IV, P. Zannetti (ed.), Elsevier Applied Science, London, pp. 387-401.

Flick, R. 1993. Modeling Management of Agricultural Ecosystems using Fuzzy Set Theory: Methodological Issues, paper presented at the joint meetings of the Western Agricultural Economics Association and the Canadian Agricultural Economics and Farm Management Society, Edmonton, Alberta.

Foster, G. R. 1987. User Requirements. USDA-Water Erosion Prediction Project (WEPP). draft 6.2. USDA-ARS National Soil Erosion Research Laboratory, Purdue, IN.

Gegov, A. E. P. M. Frank., and M. Ulieru. 1994. Decentralized control of multi-variable systems by passive decomposition. Intelligent systems engineering. Proceedings of the European on Intelligent Techniques and Soft Computing. Aachen, Germany, Vol. 3, pp. 1193-1200.

Gegov, A. E. and P. M. Frank. 1996. Decentralized control of multivariable systems by active decomposition of control laws. Int. J. Control, Vol.62, No. 4, pp.781-798.

Ghoshray, S. 1996. An efficient grey prediction model by combining fuzzy logic and genetic algorithm. In Proc. Sixth International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems (IPMU'96), pp. 727-732. Granada.

Giuclea, M., A. Agapie and F. Fagarasan. 1996. GA approach for closed loop system identification. In Proc. Fourth European Conference on Intelligent Techniques and Soft Computing (EUFIT'96), pp. 410-414. Aachen, Germany.

Guillermo, M. and S. William.1989. Forest planning and decision-making under fuzzy environments: An Overview and Illustrations, Forest Science, Vol. 35, No.2, pp. 481-502.

Gupta, M. M, J. Kiszka, and G.M. Trojan. 1986. Multi-variable structure of fuzzy control system. IEEE Trans. Syst., Man, Cybern., Vol. SCM-16, No.5. pp. 638-656.

Hansen, B. 1997. SIGMAR: A fuzzy expert system for critiquing marine forecasts. AI Applications, Vol. 11, No. 1, pp. 59-68.

Hawkins, R. H. 1978. Runoff Curve Numbers with Varying Site Moisture. ASCE, J, of Irrigation and Drainage Engineering, Vol, 104, No. IR4, pp. 389-398.

Hawkins, R. H., A. T. Hjelmfelt.,and A. W. Zevenbergen., 1985. Runoff probability, storm depth and curve numbers. ASCE, J, of Irrigation and Drainage Engineering, Vol. 111, No 4. pp. 330-340.

Huggins, L. F., and E. J. Monke, 1966. The mathematical simulation of the hydrology of small watersheds. Tech. Rep. No. 1, Water Resources Research Centre, Purdue University, West Lafayette, IN.

Huber, W.C., J. P. Heaney, and S. J. Nix. 1981. Stormwater Management Model. User's manual version III. Department of Environmental Engineering and Science, University of Florida, Gainsville, FL.

International Eelectrotechnical Committee (IEC) 1993. Industrial process: Measurement and Control Part:7- Fuzzy Control Programming pp. 1-57 Frankfourt, Germany.

Jackson, P. 1986. Introduction to Expert Systems. Addison- Wesley, Reading, MA.

Jackson, T. J., and W. J. Rawls. 1981. SCS curve numbers from a Landsat data base. Water Resources Bulletin, 17, pp. 857-862.

Kaufmann, A. 1975. Introduction to the Theory of Fuzzy Subsets, Vol. 1, Fundamental Theoretical Elements, Academic Press, New York, NY.

Kim K. C. and J. H. Kim. 1995. Evolutionary programming based multi-criteria fuzzy expert system. In Proc. Eight Australian Joint Conference on Artificial Intelligence, pp. 475-482.

Knisel, W. G. (Ed.) 1980. CREAMS: a field-scale model for chemicals, runoff and erosion from agricultural management systems. US Department of Agriculture, Conservation Research Report No. 26.

Knisel, W. G., F. M. Davis, and R. A. Leonard. 1990. GLEAMS: Version 2.0, Part III, Users Manual. USDA, Agric. Research Service, Southeast Watershed Research Laboratory, Tifton, Georgia. (Available: Williams, J. R., USDA, Grassland, Soil and Water Research Laboratory, Temple, Texas.)

Kosko, B. 1992. Neural Networks and Fuzzy Systems. Englewood Cliffs: Prentice Hall.

Koczy, L. T. and K. Hirota. 1992.Reasoning by analogy with fuzzy rules. FUZZ-IEEE, San Diego, pp. 263-270.

Koczy, L. T. and K. Hirota. 1993. Ordering, distance and closeness of fuzzy set. Fuzzy Sets Syst., Vol. 59 pp. 281-293.

Kraft D. H., F. E.Petry , B. P. Buckles, and T. Sadasivan. 1995. Applying genetic algorithms to information retrieval systems via relevance feedback. In Bosc P. and Kacprzyk J. (eds) Fuzziness in DataBase Management Systems, pp. 330-344. Physica-Verlag Heidelberg, Germany.

Kumar, S., and S. C. Jain 1982. Application of SCS Infiltration Model. AWRA, Water Resources Bulletin, Vol 18. No 3. pp. 503-507.

Lee, C. C. 1990a. Fuzzy logic in control systems: fuzzy logic controller, Part I. IEEE Transactions on Systems, Man and Cybernetics, 20, pp. 404-418.

Lee, C. C. 1990b. Fuzzy logic in control systems: fuzzy logic controller, Part II. IEEE Transactions on Systems, Man and Cybernetics, 20, pp. 419-435

Lee, S. and R. M. O'Keefe. 1994. Developing a strategy for expert system verification and validation. IEEE Transactions on Systems, Man and Cybernetics, Vol. 24. No.4, pp. 643-655.

Lehner, P. E. 1989. Toward an empirical approach to evaluating the knowledge base of an expert system. IEEE. Transactions on Fuzzy Syst. Vol. 5, No 1, pp. 658-662.

Leonard, R. A., W. G. Knisel, and A. Sull. 1987. GLEAMS: groundwater loading effects of agricultural management systems. Trans. ASAE 30. 5, pp. 1404-1418.

Linley, R. K., and N. H. Crayford, 1960. Computation of a Synthetic Stream flow Record on a Digital Computer, International Association of Scientific Hydrology Publication 51, pp. 526-538.

Linkens. D. A. and H. O. Nyongesa. 1996. A hierarchical multi-variable fuzzy controller for learning with genetic algorithm. Int. J. Control, Vol.63, No. 5, pp. 865-883.

Mamdani. E.E. 1974. Application of fuzzy systems for control of simple dynamic plant, IEEE Control Sci., Vol. 121, No. 12.

Mamdani, E. H. and S. Assilian. 1981. An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller. Fuzzy Reasoning and its Applications. Academic Press .

March, S., V. W. Huang., J. Sibigtroin., D. Paloian., D. Mazuelos., M. Leung., T. Tomazin. , S. Osborn., A. DeCastro., C. C. Hung, and T. Spielman. 1993. Fuzzy Logic Educational Program. Produced for Motorola by Cortex Communications, Inc. Austin. Texas.

Martinec, J., 1975. Snowmelt-runoff model for stream flow forecasts, Nordic hydrology 6. 3, pp. 145-154.

Mbajiorgu, C. C., 1992. Watershed Resources Management Model: a Process-Based, Continuous, Distributed-Parameter Mathematical Watershed Model Incorporating Conservation Structures. Ph.D. Thesis, Dept. of Agric. Eng. Technical University of Nova Scotia, Halifax, Nova Scotia, Canada.

McCuen, R. H., 1982. A Guide to Hydrologic Analysis using SCS Methods. Prentice-Hill, Inc. Englewood Cliffs, New Jersey.

McCuen, R. H., S. L. Wong, and W. J. Rawls. 1984. Estimation Urban Time of concentration. ASCE, Hydrolic Engineering Journal 110(7) 12,pp. 887-904.

McBratney. A. B., and A.W. Moore. 1985. Application of Fuzzy Sets to Climatic Classification, Agricultural and Forest Meteorology, 35, pp. 165-185.

McBratney. A. B., and J. J. De Gruijiter. 1992. A continuum approach to soil classification by modified fuzzy k-means with extragrads. J. Soil Sci. 43, pp. 159-175.

McCarthy, J., P. W. Abrahams., J. D. Edwards., T. P. Hart., and M. I. Levin 1965. LISP 1.5 Programmer's Manual. 2nd edn. MIT Pres. Cambridge, MA.

McNeill, F. M. 1994. Fuzzy Knowledge Builder (FKB). Fuzzy System Engineering, 12223 Wilsey Way, Poway, CA 92064.

Minsky, M., 1986. The Society of Mind. Simon and Schuster, Inc. NY.

Mockus, V. 1964. SCS National Engineering Handbook 4: Hydrology. U.S Soil Conservation Service, Washington, D.C.

Murtha, J. 1995. Applications of Fuzzy Logic in Operational Meteorology. Scientific Services and Professional Development Newsletter, Canadian Forces Weather Service, pp. 561-567.

Myers, B. A. (Ed.). 1992. Languages for Developing User Interfaces. Jones and Bartlett Publishers, Boston, MA.

Nash, J. E., 1988. Foreword. J. of Hydrology Vol. 100. Elsever Sc. Publ., Amsterdam.

Nauck, D. and Kruse. R. 1992. Interpreting changes in the fuzzy sets of a self-adaptive neural fuzzy controller. In Proc. 2nd on industrial applications of fuzzy control and intelligent systems IFIS'92 pp. 146-152. College Station, Texas

Negoita M. G. and E. Roventa. 1994. Genetic algorithms for building knowledge bases in fuzzy expert systems. In Proc. Third International Conference on Fuzzy Logic, Neural Nets and Soft Computing, pp. 455-456.Iizuka, Japan.

Newell, C., J. F. Hasbeek, P. B. Bedient. 1990. OASIS: A graphical Decision Support System for Ground water Contamination: Sue of Models in Decision Making. Kluwer Academic Publishers. P: 59-62. tliffe, E. 1976. Computer-Based medical consultations MYCIN, New York, American Elsevier.

Norwich, A. and I. B. Turksen. 1984. "A model for the measurement of membership and consequences of its empirical implementation". Fuzzy Sets Syst., Vol. 12, pp. 1-24.

Perneel C., Themlin J. M., Renders J. M., and Acheroy M. 1995. Optimization of fuzzy expert systems using genetic algorithms and neural networks. IEEE Transactions on Fuzzy Systems 3(3) pp. 300--312.

Raman, H. and N. Sunilkumar 1995. Multivariate modelling of water resources time series using artificial neural networks. Hydrological Sciences Journal. 40, 2 pp. 145-163.

Rao, V. B. and H. V. Rao (1993). C++ Neural Networks and Fuzzy Logic, MIS: Press, New York NY.

Renard, K. G., G. R. Foster, G. A. Weesies, and J. P. Porter, 1991. RUSLE: Revised universal soil loss equation. Journal of Soil and Water Conservation. 46. 1, pp. 30-33.

Ross, J. L. 1992. Intellignce artificielle en langage C. pp. 384 Masion, Paris.

Rouhani, S. and R. Kangari, 1987. Landfill site selection, Chapter 10: Expert Systems: Application to Urban Planning. Springer-Verlag Berlin Heidelberg New York.

Runkler, T. A. 1997. Selection of Appropriate Defuzzification Methods Using Application Specific Properties. IEEE Transaction on Fuzzy Syst., Vol 5, No. 1, pp. 72-79.

Schmidt, E. J., and R. E. Schulze, 1987a. Flood volume and peak discharge from small Catchments in Southern Africa, Based on the SCS Technique. Agricultural Catchments research Unit Report No. 24. Department of Agricultural Engineering University of Natal, Pietermaritzburg 3201, South Africa.

Schmidt, E. J., and Schulze, R. E. 1987b. User Manual for SCS based design runoff estimation in Southern Africa, Based on the SCS Technique. Agricultural Catchments Research Unit Report No. 25. Department of Agricultural Engineering, University of Natal, Pietermaritzburg 3201, South Africa.

Schroeder. K, A. Kamel, J. Sticklen, R. Ward, J. Ritchie, and A. Rafea, Akram Salah, 1994. A Knowldge Level Architecture for the Management of Irrigated Wheat, Agriculture and Natural Resource Management Workshop, AAAI-94, Seattle, Washington.

Schueler, T. R. 1987. Controlling Urban Runoff: A practical manual for planning and designing urban BMPs: Washington metropolitan water resources planning board, Washington Council of Govt. Washington D.C.

Schwartz, T. J. 1994. IEEE gets fuzzy. IEEE Expert. Vol. 9. No 4. pp. 72-73.

Schwartz, D. G, G. J. Klir. And Y. Ezawa 1994. Application of fuzzy sets and approximate reasoning. In IEEE, Vol. 82, No. 4, pp. 482-498.

Shakouri , A, P. Van den Bosch, H. van Nauta Lemke, and J. Dijkman. 1982. Fuzzy controller for multivariable systems. In Proc. Sec and IFAC Symp. Computer-Aided Design of Multivarable Technological Syst.

Shortliffe, E. 1976. Computer-Based Medical Consultations MYCIN, New York, American Elsevier

Shen, Q. and R. Leitch 1993. Fuzzy qualitative simulation. IEEE Transactions on Syst. Man. Cybernetics, 23(4), pp.1038-1061.

Shukla, M. B., R. Kok., S. O. Prasher., G. Clark., and R. Lacroix. 1996. Use of Artificial neural Networks in Transient Drainage Design. Tran, ASAE, Vol. 39. 1, pp. 119-124.

Skaggs, R. W. 1976. A water management model for high water table soils. ASAE Paper No1. Pp. 75-2524, ASAE St. Joseph, MI.

Smith, M. H. 1993. Parallel dynamic switching of reasoning methods in a fuzzy system. In IEEE int. Conf. Fuzzy Syst., San Francisco, CA. pp. 968-973.

Smith, J. and R. N. Eli. 1995. Neural-Networks Models of rainfall-runoff process. Journal of Water Resource Planing and Management, Vol. 121. pp. 499-508.

Soil Conservation Service (SCS) 1972. National Engineering Handbook 4: Hydrology. U.S Soil Conservation Service, Washington, D.C.

Soil Conservation Service (SCS) 1986. Urban Hydrology for Small Watersheds, TR-55.

TILShell User's Manual., 1993.Togai InfraLogic, Inc. CA.

Turksen, I. B. 1986. Interval-Valued Fuzzy Sets Based on Normal Forms. Fuzzy Sets Syst.. Vol. 20, 2, pp. 191-210.

Turksen, I. B. 1991. Measurement of membership functions and their acquisitions. Fuzzy Sets Syst.. Vol. 40, pp. 5-38.

Turksen, I. B., and Y. Tian. 1995. Two-Level Tree Search in Fuzzy expert systems. IEEE Transactions on Syst. Man. Cybernetics, 25(4), pp.555-568.

USDA, 1980. CREAMS - A field scale model for Chemicals, Runoff, and Erosion for Agricultural Management Systems. Conservation research report No. 26. P: 643 Science and Education Administration, Washington, D.C.

Valente de Oliveira. J. 1996. A design methodology for fuzzy system interfaces. IEEE Transaction on fuzzy systems. Vol. 3, No. 4, pp. 404-414.

Varas, M. A., and M. Von Chrismar. 1995. Expert System for the selection of methods to calculate design flood flows. Hydrologic Sciences Journal Vol 40, No 6. pp.739-749.

Walichiewicz. L. 1984. Decomposition of linguistic rules in the design of a multidimensional fuzzy control algorithm, in Cybernetics System. Res., Vol. 2.

Wight, J. R. (Ed.). 1983. SPUR: simulator of production and utilization of rangelands. A rangeland model for management and research. Misc. Publ. No. 1431, pp. 120. U. S. Department of Agriculture, Washington, D. C.

Williams, J. R., A. D. Nick, and J. G. Arnold. 1985. Simulator for water resources in rural basin. J. Hydraul. Eng. 111. 6, pp. 970-986.

Wilson, B. N., B. J. Barfield, and R. C. Warner. 1986. Simple models to evaluate nonpoint pollution and control. In: Agricultural Nonpoint Source Pollution. Dev. Environ. Model. 10 pp. 231-263.

Wischmeier, G. W., 1959. A rainfall erosion index for a universal soil-loss equation. Soil Science Society of America Journal. Proc. Vol. 23, pp.246-249.

Woldt. W., I. Bogardi and A. Bardossy 1992. Fuzzy geostatistical mapping of groundwater contamination in three dimensions, In: Geostatistical Methods: Recent Developments and Applications to Surface and Subsurface Hydrology A. Bardossy (ed.), UNESCO IHP Series, Paris, pp. 91-108.

Wood, M. K., and W. H. Blackburn., 1984. An evaluation of the hydrologic soil groups as used in the SCS runoff method on rangelands. Water Resources Bulletin, Vol, 20, NO. 3, pp. 379-389.

Yager, R. R., and D. P. Filev. 1994. Essentials of Fuzzy Modeling and Control, John Wiley & Sons, New York, NY, USA.

Yoon, J. and G. Padmanabhan., 1995. Heuristic Knowledge-Based Tool for rainfall Synthesis, Runoff Estimation and Hydrograph Generation. Trans, ASAE, Vol. 38. 5, pp. 1393-1403

Young, R. A., C. A. Onstad, D. D. Bosch and W.P. Anderson. 1994. Agricultural Non-Point Source pollution model, Version 4.02. AGNPS user's guide.US Department of Agriculture - Agricultural Research Service, North Central Soil Conservation Research Laboratory, North Iowa Ave., Morris, MN 56267.

Young, R. A., Onstad C. A, Bosch, D. D, and Anderson, W. P., 1994. Agricultural Non-Point Source Pollution Model. AGNPS User's Guide, USDA-ARS, Morris, Minnesota 56267.

Zadeh, L. A., 1965. Fuzzy sets. Information and Control 8, pp. 338-353.

Zadeh, A. L. 1973. Outline of a new approach to the analysis of complex systems and decision processes, IEEE Trans. Syst., Man, Cybern., Vol. SCM-1, No.1.

Zadeh, A. L. 1992. The Calculus of Fuzzy If-Then Rules, AI Expert, Vol. 7(3) pp. 22 -27.

Zadeh, L. A. and J. Kacprzyk, ed. 1992. Fuzzy Logic for the Management of Uncertainty, John Wiley and Sons, New York NY.

Zadeh, L. A. 1996. Fuzzy logic = Computing with words. IEEE Transaction on Fuzzy and Syst. Vol. 4(2) pp. 103-111.

Zeng, X. L., and M. G. Singh. 1995. Approximation theory of fuzzy systems -MISO case. IEEE Transaction on Fuzzy Syst. Vol. 3(2) pp. 219-253.

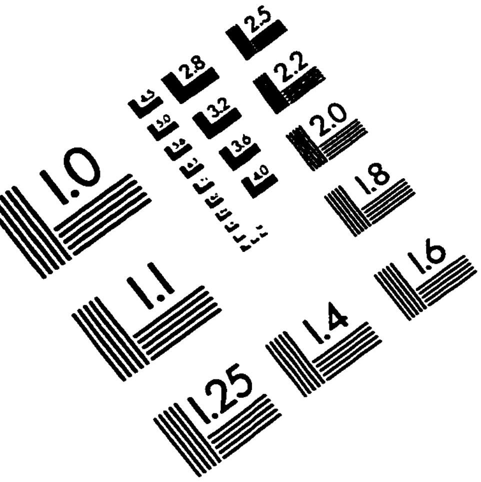
Zhang L., Wang L., Zang Y., Seki H., and Itoh H.1995. On rule checking and learning in an acupuncture diagnosis fuzzy expert system by genetic algorithm. In Proc. Fourth IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'95), pp. 455-460. Yokohama.

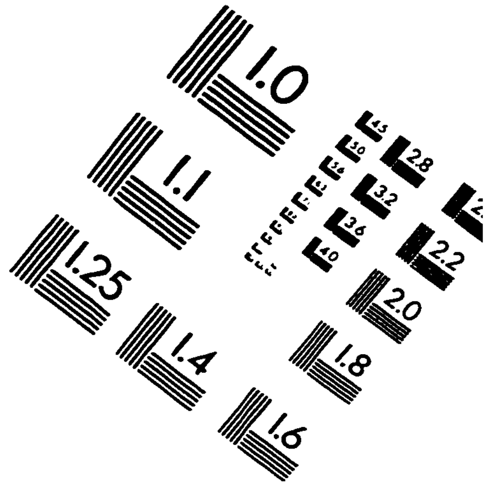
Zimmermann, H. J. 1987. Fuzzy Sets, Decision Making and Expert Systems. Kluwer-Academic Publishers, Boston, MA.

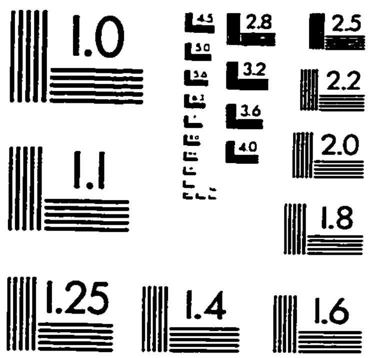
Zimmermann, H. J. 1992. Fuzzy Set theory and its Applications. Kluwer-Nijhoff Publishers, Boston. MA.

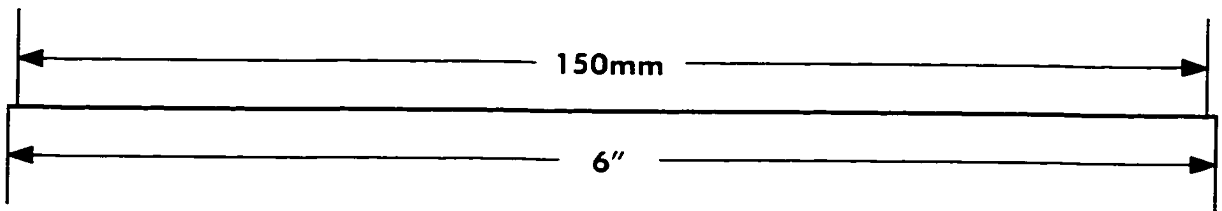
Zimmermann H. J. 1994. Hybrid approaches for fuzzy data analysis and configuration using genetic algorithms and evolutionary methods. In Zurada J. M., Marks II R. J., and Robinson C. J. (eds) Computational Intelligence Imitating Life, pp. 364-370. IEEE Press, New York.

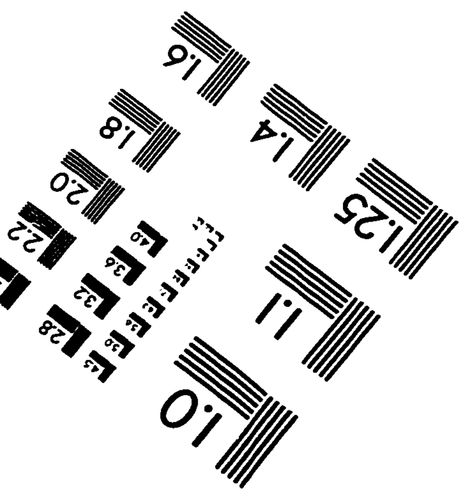
IMAGE EVALUATION TEST TARGET (QA-3)

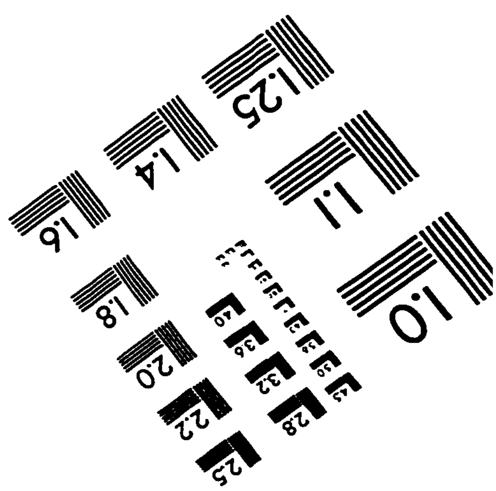












APPLIED IMAGE . Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989