Tutorial on Fuzzy Logic Applications in Power Systems

Prepared for the IEEE-PES Winter Meeting in Singapore
January, 2000

Edited by:

K. Tomsovic, Washington State University
M.Y. Chow, North Carolina State University

Contributors:

B. Baer, Seimens-Empros
K. Butler, Texas A&M University
T. Hiyama, Kumamoto University
J. Kubokawa, Hiroshima University
K.Y. Lee, Pennsylvania State University
P. Luh, University of Conneticut
D. Niebur, Drexel University
O.P. Malik, University of Calgary
J. Momoh, Howard University
H. Mori, Hiroshima University
H. Sasaki, Hiroshima University
D. Srinivasan, National University of Singapore
G L. Torres, Escola Federal de Engenharia de Itajuba
R. Yokoyama, Tokyo Metropolitan University
J. Zhu, North Carolina State University

Contributors to Tutorial on Fuzzy Logic Applications in Power Systems by Chapter

Chapter 1:
M.Y. Chow, North Carolina State University
K. Tomsovic, Washington State University
Chapter 2:
K. Butler, Texas A&M University
G L. Torres, Escola Federal de Engenharia de Itajuba
Chapter 3:
O.P. Malik, University of Calgary
D. Niebur, Drexel University
T. Hiyama, Kumamoto University
Chapter 4:
K.Y. Lee, Pennsylvania State University
Chapter 5:
J. Momoh, Howard University
D. Srinivasan, National University of Singapore
K. Tomsovic, Washington State University
B. Baer, Seimens-Empros
Chapter 6:
M.Y. Chow, North Carolina State University
P. Luh, University of Conneticut
S. Lin, University of Conneticut
J. Zhu, North Carolina State University
Chapter 7:
H. Sasaki, Hiroshima University
R. Yokoyama, Tokyo Metropolitan University
J. Kubokawa, Hiroshima University
Chapter 8:
H. Mori, Meiji University

Table of Contents

Chapter 1 Overview

This tutorial provides attendees with a comprehensive overview of fuzzy logic applications in power systems. Every effort was made to ensure the material was self-contained and requires no specific experience in fuzzy logic methods. At the same time, this booklet includes contributions, which are undoubtedly state-of-the-art research. Thus, it is hoped that practitioners at all levels will find useful information here. Fuzzy logic technology has achieved impressive success in diverse engineering applications ranging from mass market consumer products to sophisticated decision and control problems [1]. Applications within power systems are extensive with more than 100 archival publications in a recent survey [2,3]. Several of these applications have found their way into practice and fuzzy logic methods are becoming another important approach for practicing engineers to consider .

In 1965, L.A. Zadeh laid the foundations of fuzzy set theory [4] as a method to deal with the imprecision of practical systems. Bellman and Zadeh write: "Much of the decision-making in the real world takes place in an environment in which the goals, the constraints and the consequences of possible actions are not known precisely" [5]. This "imprecision" or fuzziness is the core of fuzzy sets or fuzzy logic applications. Fuzzy sets were proposed as a generalization of conventional set theory. Partially as result of this fact, fuzzy logic remained the purview of highly specialized and mathematical technical journals for many years. This changed abruptly with the highly visible success of several control applications in the late 1980s.

Heuristics, intuition, expert knowledge, experience, and linguistic descriptions are obviously important to power engineers. Virtually any practical engineering problem requires some “imprecision” in the problem formulation and subsequent analysis. For example, distribution system planners rely on spatial load forecasting simulation programs to provide information for a variety of planning scenarios [6]. Linguistic descriptions of growth patterns, such as close by or fast, and design objectives, such as, prefer or reduce, are imprecise in nature. The conventional engineering formulations do not capture such linguistic and heuristic knowledge in an effective manner.

Fuzzy logic implements human experiences and preferences via membership functions and fuzzy rules. Fuzzy membership functions can have different shapes depending on the designer's preference and/or experience. The fuzzy rules, which describe relationships at a high level (in a linguistic sense), are typically written as antecedent-consequent pairs of IF-THEN statements. Basically, there are four approaches to the developing fuzzy rules [7]: (1) extract from expert experience and control engineering knowledge, (2) observe the behavior of human operators, (3) use a fuzzy model of a process, and (4) learn relationships through experience or simulation with a learning process. These approaches do not have to be mutually exclusive. Due to the use of linguistic variables and fuzzy rules, the system can be made understandable to a non-expert operator. In this way, fuzzy logic can be used as a general methodology to incorporate knowledge, heuristics or theory into controllers and decision-makers.

This tutorial begins with a general section on fuzzy logic techniques and methods. Simplified examples are used to highlight the fundamental methodologies. Control applications are addressed in chapters 3 and 4. Chapter 3 provides fundamental analysis as well as a brief description of a controller in field use. Chapter 4 presents more advanced concepts, including both control design and stability analysis, useful for the more experienced developer. Approaches based on approximate reasoning in expert systems are presented in Chapter 5, with a specific application to diagnostic systems. This is followed by two extensive chapters on optimization problems. Chapter 6 presents applications in spatial load forecasting and in scheduling. Applications on generation expansion planning and optimal power flow in Chapter 7 highlight an alternative approach to optimization The tutorial concludes with a chapter on advanced applications including hybrid applications of neural nets and fuzzy logic.

A References

[1] M.Y. Chow, “Fuzzy Systems,” in *CRC Press Industrial Electronics Handbook*, D. Irwin, Ed.: CRC, 1996.

[2] J. Zhu and M.Y. Chow, “A Review of Emerging Techniques on Generation Expansion Planning,” *IEEE Transactions on Power Systems*, in press.

[3] J.A. Momoh, X.W. Ma and K. Tomsovic, “Overview and Literature Survey of Fuzzy Set Theory in Power Systems”, *IEEE Transactions on Power Systems*, Vol. 10, No.3, Aug. 1995.

[4] L. A. Zadeh, “Fuzzy Sets,” in *Information and Control*, vol. 8. New York: Academic Press, 1965, pp. 338-353.

[5] R. E. Bellman and L. A. Zadeh, “Decision-making in a fuzzy environment,” *Management Science*, vol. 17, pp. 141-164, 1970.

[6] M.Y. Chow and H. Tram, “Applications of Fuzzy Logic Technology for Spactial Load Forecasting,” *IEEE Transactions on Power Systems*, 1996, in press.

[7] M.Y. Chow and A. Menozzi, “Design Methodology of an Intelligent Controller Using Artificial Neural Networks,” presented at IECON'93, Maui, Hawaii, 1993.

Chapter 2 Fuzzy Set Fundamentals

A Fuzzy Sets

Zadeh makes a case that humans reason not in terms of discrete symbols and numbers, but in terms of fuzzy sets [1]. These fuzzy terms define general categories, but not rigid, fixed collections. The transition from one category-concept, idea, or problem state-to the next is gradual with some states having greater or less membership in the one set and then another. From this idea of elastic sets, Zadeh proposed the concept of a fuzzy set. Fuzzy sets are functions that map a value that might be a member of the set to a number between zero and one indicating its actual degree of membership. A degree of zero means that the value is not in the set, and a degree of one means that the value is completely representative of the set. This produces a curve across the members of the set. There are many books that have been written on the subject of fuzzy sets since Zadeh introduced the fuzzy set concept in 1965 [1-19].

A.1 Membership Functions – Fundamental Definitions

Let X be a set of objects, called the universe, whose elements are denoted x. Membership in a subset A of X is the membership function, μ_A from X to the real interval [0,1]. The universe is all the possible elements of concern in the particular context. A is called a fuzzy set and is a subset of X that has no sharp boundary. μ_A is the grade of membership x in A. The closer the value of μ_A is to 1, the more x belongs to A. The total allowable universe of values is called the *domain* of the fuzzy set. The domain is a set of real numbers, increasing monotonically from left to right where the values can be both positive and negative. A is completely characterized by the set of pairs

$$A = \{(x, \mu_A(x)), x \in X)\} \tag{1}$$

Support of a fuzzy set A in the universal set X is the crisp set that contains all the elements of X that have a nonzero membership grade in A. That is

$$\text{supp } A = \left\{x \in X \mid \mu_A(x) > 0\right\} \tag{2}$$

With a finite support, we'll let x_i be an element of the support of fuzzy set A and that μ_i a grade of membership in A. Then A is written by convention as

$$A = \frac{\mu_1}{x_1} + \ldots + \frac{\mu_n}{x_n} = \sum_{i=1}^{n} \frac{\mu_i}{x_i} \tag{3}$$

When X is an interval of real numbers, a fuzzy set A is expressed as

$$A = \int_X \frac{\mu_A(x)}{x} \tag{4}$$

An *empty* fuzzy set has an empty support which implies that the membership function assigns 0 to all elements of the universal set.
A technical concept closely related to the support set is the alpha-level set or the "α-*cut*". An alpha level is a threshold restriction on the domain of the fuzzy set based on the membership grade of each domain value. This set, $A_•$, is the α-cut of A which contains all the domain values that are part of the fuzzy set at a minimum membership value of α. There are two kinds of α-cuts: weak and strong. The weak α-cut is defined as $A_\alpha = \{x \in X, \mu_A(x) \geq \alpha\}$ and the strong •-cut as $A_a = \{x \in X, \mu_A(x) > a\}$. Also, the alpha-level set describes a power or strength function that is used by fuzzy models to decide whether or not a truth value should be considered equivalent to zero. This is an important facility that controls the execution of fuzzy rules as well as the intersection of multiple fuzzy sets.

The degree of membership is known as the membership or truth function since it establishes a one-to-one correspondence between an element in the domain and a truth value indicating its degree of membership in the set. It takes the form,

$$\mu_A(x) \leftarrow f(x \in A) \tag{5}$$

The triangular membership function is the most frequently used function and the most practical, but other shapes are also used. One is the trapezoid which contains more information than the triangle. A fuzzy set can also be represented by a quadratic equation (involving squares, n^2, or numbers to the second power) to produce a continuous curve. Three additional shapes which are named for their appearance are: the S-function, the pi-function, and the Z-function.

A.2 Set Operations

Union and Intersection of Fuzzy Sets

The classical union $(\cup)$ and intersection $(\cap)$ of ordinary subsets of X are extended by the following formulas for intersection, $A \cap B$, and union, $A \cup B$:

$$\forall\, x \in X,\ \mu_{A\cup B}(x) = \max(\mu_A(x),\ \mu_B(x)) \qquad (6)$$

$$\forall\, x \in X,\ \mu_{A\cap B}(x) = \min(\mu_A(x),\ \mu_B(x)) \qquad (7)$$

where $\mu_{A\cup B}$ and $\mu_{A\cap B}$ are respectively the membership functions of $A\cup B$ and $A\cap B$.

For each element x in the universal set, the function in (6) takes as its argument the pair consisting of the element's membership grades in set A and in set B and yields the membership grade of the element in the set constituting the union of A and B. The disjunction or union of two sets means that any element belonging to either of the sets is included in the partnership which expresses the maximum value for the two fuzzy sets involved.

The argument to the function in (7) returns the membership grade of the element in the set consisting of the intersection of A and B. A conjunction or intersection makes use of only those aspects of Set A and Set B that appear in both sets which expresses the minimum value for the two fuzzy sets involved.

<u>Complement of a Fuzzy Set</u>

The complement of A, $\sim A$, which is the part of the domain not in a set, can also be characterized by *Not-A*. This is produced by inverting the truth function along each point of the fuzzy set and is defined by the membership function

$$\forall\, x \in X,\ \mu_{\overline{A}}(x) = 1-\mu_A(x) \qquad (8)$$

The complement registers the degree to which an element is complementary to the underlying fuzzy set concept. That is, how compatible is an element's value [x] with the assertion, x is NOT y, where x is an element from the domain and y is a fuzzy region. A fuzzy complement is actually a metric. It measures the distance between two points in the fuzzy regions at the same domain. The linear displacement between the complementary regions of the fuzzy regions determines the degree to which one set is a counter example of the other set. We can also view this as a measure of the fuzziness or information entropy in the set.

A.3 Defining Fuzzy Sets

The steps below give general guidelines in defining fuzzy sets [1].

(i) *Determine the type of fuzzy measurement*. Fuzzy sets can define

- orthogonal mappings between domain values and their membership in the set ("ordinary fuzzy set")
- differential surfaces which represent the first derivative of some action, degree of change between model states, or the force of control that must be applied to bring a system back to equilibrium
- a proportional metric which reflects a degree of proportional compatibility between a control state and a solution state
- a proportionality set which reflects a degree of proportionality between a control state and a solution state

(ii) *Choose the shape (or surface morphology) of the fuzzy set*. The shape maps the underlying domain back to the set membership through a correspondence between the data and the underlying concepts. Some possible shapes are triangular, trapezoidal, PI-curve, bell-shaped, S-curves, and linear. Every base fuzzy set must be normal.

(iii) *Select an appropriate degree of overlap*. The series of individual fuzzy sets, associated with the same solution variable, are converted into one continuous and smooth surface by overlapping each fuzzy set with its neighboring set. The degree of overlap depends on the concept modeled and the intrinsic degree of imprecision associated with the two neighboring states.

(iv) *Ensure that the domains among the fuzzy sets associated with the same solution variables share the same universe of discourse.*

B Expert Reasoning and Approximate Reasoning

B.1 Fuzzy Measures

The fuzzy measure assigns a value to each crisp set of the universal set signifying the degree of evidence or belief that a particular element belongs in the set. For example, we might want to diagnose an ill patient by determining whether this patient belongs to the set of people with, pneumonia, bronchitis, emphysema, or a common cold. A physical examination may provide us with helpful yet inconclusive evidence. Therefore, we might assign a high value, 0.75, to our best guess, bronchitis, and a lower value to the other possibilities, such as .45 to pneumonia, .3 to a common cold, and 0 to emphysema. These values reflect the degree to which the patient's symptoms provide evidence for one disease rather than another, and the collection of these values constitutes a fuzzy measure representing the uncertainty or ambiguity associated with several well-defined alternatives.

A fuzzy measure is a function [7]

$$g : \mathsf{B} \rightarrow [0,1],$$

where $\mathrm{B} \subset \mathrm{P}(X)$, called a Borel field, is a family of subsets of X such that:

1. $\varnothing \in \mathrm{B}$ *and* $X \in \mathrm{B}$;

2. If $A \in \mathrm{B}$, *then* $\overline{A} \in \mathrm{B}$

3 It is closed under the operation of set union, that is, if $A \in \mathrm{B}$ and $B \in \mathrm{B}$, then also $A \cup B \in \mathrm{B}$.

Since $A \cup B \supseteq A$ and $A \cup B \supseteq B$, then due to monotonicity we have $\max\left[g(A), g(B)\right] \le g(A \cup B)$. Similarly since $A \cap B \subseteq A$ and $A \cap B \subseteq B$, $g(A \cap B) \le \min\left[g(A), g(B)\right]$.

Two large classes of fuzzy measures are referred to as belief and plausibility measures which are complementary (or dual) in the sense that one of them can be uniquely derived from the other. Given a basic assignment *m*, a belief measure and a plausibility measure are uniquely determined by the formulas

$$Bel(A) = \sum_{B \subseteq A} m(B) \quad \text{and} \quad Pl(A) = \sum_{B \cap A \neq \varnothing} m(B)$$

which are applicable for all $A \in \mathrm{B}(X)$. Also *m(A)* refers to the degree of evidence or belief that a specific element of *X* belongs to the set *A* alone. The *belief measure, Bel(A)*, represents the total evidence or belief that the element belongs to the set *A* as well as to the various special subsets of A. The *plausibility measure, Pl(A)*, represents not only the total evidence or belief that the element in question belongs to set *A* or to any of its subsets but also the additional evidence or belief associated with sets that overlap with *A*. There are also three important special types of plausibility and belief measures, *probability measures* and a pair of complementary measures referred to as *possibility* and *necessity* measures [7].

B.2 Approximate Reasoning

The root mechanism in a fuzzy model is the proposition. These are statements of relationships between mode variables and one or more fuzzy regions. A series of conditional and unconditional fuzzy associations or propositions is evaluated for its degree of truth and all those that have some truth contribute to the final output state of the solution variable set. The functional tie between the degree of truth in related fuzzy regions is called the method of implication. The functional tie between fuzzy regions and the expected value of a set point is called the method of defuzzification. Taken together these constitute the backbone of approximate reasoning. Hence an approximate reasoning system combines the attributes of conditional and unconditional fuzzy propositions, correlation methods, implication (truth transfer) techniques, proposition aggregation, and defuzzification [1].

Unlike conventional expert systems where statements are executed serially, the principal reasoning protocol behind fuzzy logic is a parallel paradigm. In conventional knowledge-based systems pruning algorithms and heuristics are applied to reduce the number of rules examined, but in a fuzzy system all the rules are fired.

B.3 The Role of Linguistic Variables

Fuzzy models manipulate linguistic variables. A linguistic variable is the representation of a fuzzy space which is essentially a fuzzy set derived from the evaluation of the linguistic variable. A linguistic variable encapsulates the properties of approximate or imprecise concepts in a systematic and computationally useful way.

The organization of a linguistic variable is

$$L_{\mathrm{var}} \leftarrow \{q_1 \ldots q_n\}\,\{h_1 \ldots h_n\}\ fs \qquad (9)$$

where predicate *q* represents frequency qualifiers, *h* represents a hedge and *fs* is the core fuzzy set. The presence of qualifier(s) and hedge(s) are optional. Hedges change the shape of fuzzy sets in predictable ways and function in the same fashion as adverbs and adjectives in the English language. Frequency and usuality qualifiers reduce the derived fuzzy set by restricting the truth membership function to a range consistent with the intentional meaning of the qualifier. Although a linguistic variable may consist of many separate terms, it is considered a single entity in the fuzzy proposition.

B.4 Fuzzy Propositions

A fuzzy model consists of a series of conditional and unconditional fuzzy propositions. A proposition or statement establishes a relationship between a value in the underlying domain and a fuzzy space. A *conditional fuzzy proposition* is one that is qualified as an *if* statement. The proposition following the *if* term is the antecedent or predicate and is an arbitrary fuzzy proposition. The proposition following the *then* term is the consequent and is also any arbitrary fuzzy proposition.

$$\text{If } w \text{ is } Z \text{ then } x \text{ is } Y$$

interpreted as

x is a member of *Y* to the degree that *w* is a member of *Z*

An *unconditional fuzzy proposition* is one that is not qualified by an *if* statement.

x is *Y*

where *x* is a scalar from the domain and *Y* is a linguistic variable. Unconditional statements are always applied within the model and depending on how they are applied, serve either to restrict the output space or to define a default solution space. We interpret an unconditional fuzzy proposition as

X is the minimum subset of *Y*

when the output fuzzy set *X* is empty, then *X* is restricted to *Y*, otherwise, for the domain of *Y*, *X* becomes the *min(X,Y)*. The solution fuzzy space is updated by taking the intersection of the solution set and the target fuzzy set.

If a model contains a mixture of conditional or unconditional propositions, then the order of execution becomes important. Unconditional propositions are generally used to establish the default support set for a model. If none of the conditional rules executes, then a value for the solution variable is determined from the space bounded by the unconditionals. For this reason, they must be executed before any of the conditionals.

The effect of evaluating a fuzzy proposition is a degree or grade of membership derived from the transfer function,

$$\mu \leftarrow \left(x \in Y\right) \tag{10}$$

where *x* is a scalar from the domain and *Y* is a linguistic variable. This is the essence of an approximate statement. The derived truth membership value establishes a compatibility between *x* and the generated fuzzy space *Y*. This truth value is used in the correlation and implication transfer functions to create or update fuzzy solution space. The final solution fuzzy space is created by aggregating the collection of correlated fuzzy propositions.

B.5 Fuzzy Implication

The *monotonic* method is a basic fuzzy implication technique for linking the truth of two general fuzzy regions. When two fuzzy regions are related through a simple proportional implication function,

if *x* is *Y* then *z* is *W*

functionally represented by the transfer function,

$$z = f\left(\left(x, Y\right), W\right) \tag{11}$$

then under a restricted set of circumstances, a fuzzy reasoning system can develop an expected value without going through composition and decomposition. The value of the output is estimated directly from a corresponding truth membership grade in the antecedent fuzzy regions. While the antecedent fuzzy expression might be complex, the solution is not produced by any formal method of defuzzification, but by a direct slicing of the consequent fuzzy set at the antecedent's truth level. Monotonic reasoning acts as a proportional correlating function between two general fuzzy regions. The important restriction on monotonic reasoning is its requirement that the output for the model be a single fuzzy variable controlled by a single fuzzy rule (with an arbitrarily complex predicate).

Implication space generated by the general *compositional rules of inference* is derived from the aggregated and correlated fuzzy spaces produced by the interaction of many statements. In effect all the propositions are run in parallel to create an output space that contains information from all the propositions. Each conditional proposition whose evaluated predicate truth is above the current alpha-cut threshold contributes to the shape of the output solution variable's fuzzy representation. There are two principal methods of inference in fuzzy systems: the min-max method and the fuzzy additive method. These methods differ in the way they update the solution variable's output fuzzy representation.

For the *min-max inference* method the consequent fuzzy region is restricted to the minimum of the predicate truth. The output fuzzy region is updated by taking the maximum of these minimized fuzzy sets. The consequent fuzzy set is modified before it is used to set each truth function element to the minimum of either the truth function or truth of the proposition's predicate. The solution fuzzy set is updated by taking, for each truth function value, the maximum of either the truth value of the solution fuzzy set or the fuzzy set that was correlated to produce the minimum of consequent. These steps result in reducing the strength of the fuzzy set output to equal to the maximum truth of the predicate and then, using this modified fuzzy region, applying it to the output by using the *OR* (union) operator. When all the propositions have been evaluated, the output contains a fuzzy set that reflects the contribution from each proposition.

The *fuzzy additive compositional inference* method updates the solution variable's fuzzy region in a slightly different manner. The consequent fuzzy region is still reduced by the maximum truth value of the predicate, but the output fuzzy region is updated by a different rule, the bounded-sum operation. Instead of taking the $\max\left(\mu_A\left[x_i\right], \mu_B\left[y_i\right]\right)$ at each point along the output fuzzy set, the truth membership functions are added. The addition is bounded by [1,0] so that the result of any addition cannot exceed the maximum truth value of a fuzzy set. The use of the fuzzy additive implication method can provide a better representation of the problem state than systems that rely solely on the min-max inference scheme.

B.6 Correlation Methods

The process of correlating the consequent with the truth of the predicate stems from the observation that the truth of the fuzzy action cannot be any greater than the truth of the proposition's premise. There are two principal methods of restricting the height of the consequent fuzzy set: correlation minimum and correlation product. The most common method of correlating the consequent with the premise truth truncates the consequent fuzzy region at the truth of the premise. This is called correlation minimum, since the fuzzy set is minimized by truncating it at the maximum of the predicate's truth. The *correlation minimum* mechanism usually creates a plateau since the top of the fuzzy region is sliced by the predicate truth value. This introduces a certain amount of information loss. If the truncated fuzzy set is multimodal or otherwise irregular, the surface topology above the predicate truth level is discarded. The correlation method, however, is often preferred over the correlation product (which does preserve the shape of the fuzzy region) since it intuitively reduces the truth of the consequent by the maximum truth of the predicate, involves less complex and faster arithmetic, and generally generates an aggregated output surface that is easier to defuzzify using the conventional techniques of composite moments (centroid) or composite maximum (center of maximum height).

While correlation minimum is the most frequently used technique, correlation product offers an alternative and, in many ways, better method of achieving the correlation. With *correlation product*, the intermediate fuzzy region is scaled instead of truncated. The truth membership function is scaled using the truth of the predicate. This has the effect of shrinking the fuzzy region while still retaining the original shape of the fuzzy set. The correlation product mechanism does not introduce plateaus into the output fuzzy region, although it does increase the irregularity of the fuzzy region and could affect the results obtained from composite moments or composite maximum defuzzification. This lack of explicit truncation has the consequence of generally reducing information loss. If the intermediate fuzzy set is multimodal, irregular, or bifurcated in other ways this surface topology will be retained when the final fuzzy region is aggregated with the output variable's undergeneration fuzzy set.

B.7 Aggregation

The evaluation of the model propositions is handled through an *aggregation* process that produces the final fuzzy regions for each solution variable. This region is then decomposed using one of the defuzzification methods.

B.8 Methods of Defuzzification

Using the general rules of fuzzy inference, the evaluation of a proposition produces one fuzzy set associated with each model solution variable. *Defuzzification* or decomposition involves finding a value that best represents the information contained in the fuzzy set. The deffuzification process yields the expected value of the variable for a particular execution of a fuzzy model. In fuzzy models, there are several methods of defuzzification that describe the ways we can derive an expected value for the final fuzzy state space.

Defuzzification means dropping a "plumb line" to some point on the underlying domain. At the point where this line crosses the domain axis, the expected value of the fuzzy set is read. Underlying all the defuzzification functions is the process of finding the best place along the surface of the fuzzy set to drop this line. This generally means that defuzzification algorithms are a compromise with or a trade-off between the need to find a single point result and the loss of information such a process entails.

The two most frequently used defuzzification methods are composite moments (centroid) and composite maximum. The *centroid* or center of gravity technique finds the balance point of the solution fuzzy region by calculating the weighted mean of the fuzzy region. Arithmetically, for fuzzy solution region *A*, this is formulated as

$$\Re \leftarrow \frac{\sum_{i=0}^{n} d_i \ \mu_A \ \left(d_i\right)}{\sum_{i=0}^{n} \mu_A \left(d_i\right)} \tag{12}$$

where d is the *ith* domain value and $\mu(d)$ is the truth membership value for that domain point. A centroid or composite moments defuzzification finds a point representing the fuzzy set's center of gravity.

A *maximum decomposition* finds the domain point with the maximum truth. There are three closely related kinds of composite maximum techniques: the average maximum, the center of maximums, and the simple composite maximum. If this point is ambiguous (that is, it lies along a plateau), then these methods employ a conflict resolution approach such as averaging the values or finding the center of the plateau.

Also there are other techniques for decomposing a fuzzy set into an expected value. The *average of maximum value* defuzzification finds the mean maximum value of the fuzzy region. If this is a single point, then this value is returned; otherwise, the value of the plateau is calculated and returned. The *average of the nonzero* region is the same as taking the average of the support set for the output fuzzy region. The *far and near edge of the support set* technique selects the value at the right fuzzy set edge and is of most use when the output fuzzy region is structured as a single-edge plateau. The *center of maximums* technique, in a multimodal or multiplateau fuzzy region , finds the highest plateau and then the next highest plateau. The midpoint between the centers of these plateaus is selected.

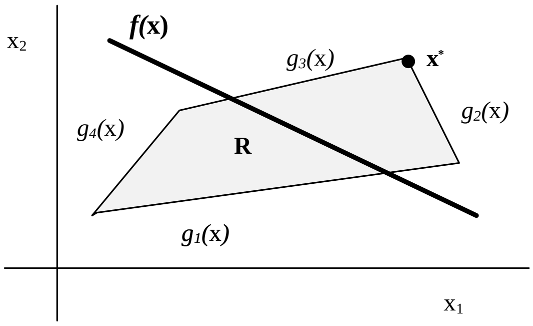
C Fuzzy Optimization

C.1 Initial Considerations

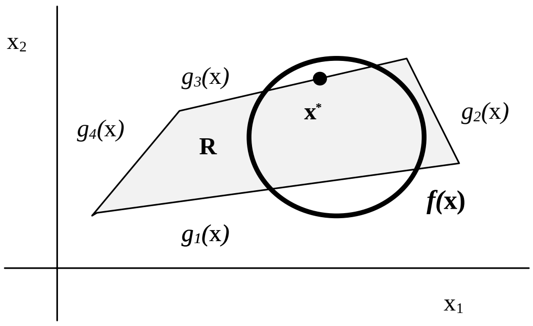
In the real world, it is not an easy task to find a useful optimal solution of a given problem because many constraints and limitations must be taken into account during this process. Usually, only the most important constraints and limitations are chosen to be used during the solution search process. Another problem is that the solution may not be unique and it depends directly on the weight of each constraint. Hence, many optimization processes have been developed in the last decades to achieve the best solution in this search process. In addition, one of the most important points in this process is the computational cost to find the best solution. Sometimes, this cost may comprise the use of a technique to solve a problem. Number of constraints, number of variables (involved to describe the problem) and poor convergence speed are some examples of computational cost. In other words, the computational problems are related to hardware processor speed, memory capacity and numerical techniques. However, the highly fast evolution of the computational world (hardware and software) allows optimization techniques, that could not be used before to solve a specific problem, to be applied successfully now. Specifically for power system problems, decomposition techniques, partitioning techniques and parallel processing are examples of recent evolution of computational techniques.

In many power system problems, the use of optimization techniques has been important to reduce costs and losses of the system. Unit commitment, economic dispatching, and optimal power flow are some areas where these techniques have been extensively used. For example, minimization of active power losses is one of the biggest challenges for power control operators. The achievement of this goal in real-time is a critical task. A possible solution for this problem is to use the Dantzig and Wolfe decomposition algorithm to partitioning the power system in many subsystems according to a geographic basis. The optimization process is applied to these subsystems, and the constraints are limited to local constraints and coupling bus constraints.

An optimization process can be defined as a maximization (or minimization) of an objective function, $f(\mathrm{x})$, subject to constraints of the problem, $g(\mathrm{x})$. These constraints define a feasible region **R**, i.e., a region that contains possible solutions of the problem. Two popular techniques have been developed for optimization process, they are linear programming and quadratic programming. Examples of these techniques, for two variables x_1 and x_2, are shown in Figure 1, where there are 4 linear inequality constraints, $g_i(\mathrm{x})$, that define the feasible region **R**, and the optimal solution is denoted by x^*.



(a) Linear Programming.



(b) Quadratic Programming

Fig.1 - Example of Techniques.

It can be verified that in the linear programming the optimal solution occurs always at an extreme point (a “corner” point) of the feasible region; while in the quadratic programming this solution can be located on the interior or boundary of the feasible region.

The two major drawbacks of these current optimization methods are: speed/convergence problems and correct representation of constraints. Usually, methods with fast speed present poor convergence, while slower methods have less convergence problems. On one hand, for example, Newton-Raphson (or other parallel tangent methods) presents a very good answer when the starting point is near the solution point; however, this methods performance can be very dependent on the shape of the involved functions. On the other hand, bisection methods (e.g., Fibonacci, cubic, and quadratic searches) are slower than tangent methods but they are more reliable. Hybrid system schemes have been proposed. Initially, the procedure starts with a bisection method until the vicinity of the optimal point; then, the procedure changes the method to a parallel tangent method.

The second drawback, correct representation of constraints, is related to the difficulty to evaluate the correct value to be incorporated in the constraint equations. Sometimes, these constraints are not well-defined by crisp functions, and the use of fuzzy values is recommended. Many fuzzy optimization methods have been proposed in the literature, where they can be classified according to the introduction of fuzzy set theory in: (a) representation of the constraints, and (b) solution method. A typical fuzzy optimization process is described in the next sections.

The main applications of fuzzy optimization in power system problems are: expansion planning [21-25], maintenance scheduling [26,27], unit commitment [28], multi-objective coordination [29-31], and optimal power flow [32-34].

C.2 Fuzzy Optimization by Pseudogoal Function

Description of the Process

Usually, optimization problems with a single-real variable are solved using bisection methods, where the main idea is to reduce an initial interval until a required minimum. Different from the classical optimization methods, the main idea in fuzzy optimization is to optimize objective function and constraints, simultaneously. In order to determine the optimal point (solution point), both objective function and constraints must be characterize by membership functions and they must be linked by a linguistic conjunction: "and" (for maximization) and "or" (for minimization).

Fuzzy optimization by pseudogoal was proposed by Bellman and Zadeh [41] and the main idea is to satisfy a fuzzy objective function and fuzzy constraints that receive the same treatment, i.e., there is no difference among the objective function and constraints. The first step is a fuzzification process of the objective function, this procedure converts the objective function $f(\mathrm{x_j})$ into a pseudogoal $F(\mathrm{x_j})$ by the following fuzzification process

$$\mu_F(\mathrm{x_j}) = \frac{f(\mathrm{x_j}) - \mathrm{I}}{\mathrm{S - I}} \qquad (13)$$

where S and I are the maximum and minimum possible values in the feasible interval for the function $f(\mathrm{x_j})$, respectively.

The constraints may also receive the same fuzzification process as above, or they are previously defined as a membership function. In the latter possibility, this definition can represent an expertise (or a linguistic value). For example, using a crisp function, a possible constraint can be $\mathrm{x} \le 3$. The same constraint can be expressed as "the good value is equal or less than 3". A possible complement of this statement may be "it is also acceptable for a value to be not so larger than 3". A possible membership function to represent this linguistic value can be

$$C(\mathrm{x}) = \begin{cases} 1 & \text{for x} \le 3 \\ a + \dfrac{k}{\mathrm{x}} & \text{for x} > 3 \end{cases}$$

where k represents how acceptable is the value larger than 3. If the value of k is small (usually, $k < 1$), only values very close to 3 are acceptable; otherwise ($k > 1$), k can represent bigger values for the membership functions. Figure 2 presents an example of these values. The constant a is only a parameter for level adjustment and it is used to turn the membership function to a continuous one.

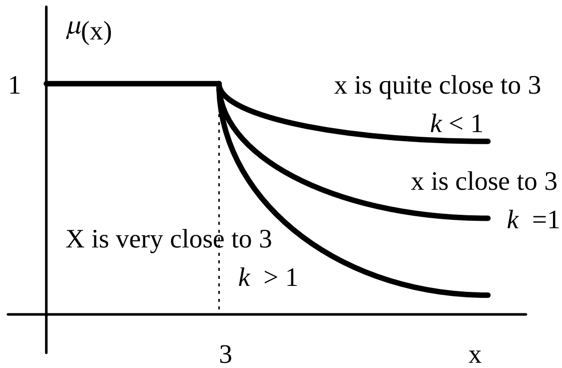


Fig. 2. - Possible Membership Functions for a Generic Constraints.

Another usual procedure is the use of fuzzy numbers to define constraints. In classical optimization, intervals define the region to be explored. In fuzzy optimization, this region can be expressed using fuzzy numbers. An example of this procedure is shown in Figure 3, where δ_1 and δ_2 can be defined as the fulfillment (or relaxation) of the constraint.

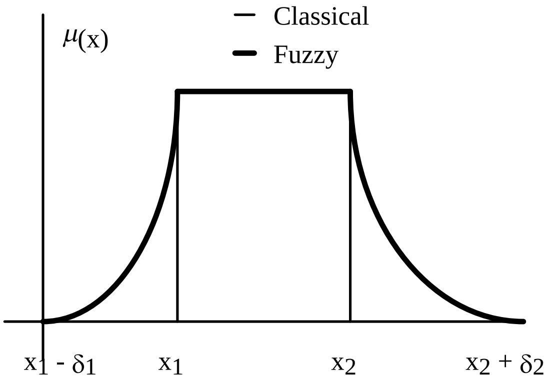


Fig. 3. - Classical and Fuzzy Intervals: $[x_1,x_2]$ and $[x_1 + \delta_1, x_2 + \delta_2]$, respectively.

After the fuzzification process, the membership of the optimal function can be found by the aggregation of all constraints and the pseudogoal. In the computation of the fuzzy maximum function, all membership functions are initially merged by the conjunction "and" (intersection of all function, operator: minimum) and then the optimal value (solution) x^* is computed by the operator maximum (i.e., the maximum-minimum value of the membership function). This procedure can be presented by the following sequence, where $G(\mathrm{x})$ represents the decision function, and $\mu_G(\mathrm{x})$ is its associated membership,

$$\mu_G(\mathrm{x}) = min(C, F)$$
$$\mathrm{x}^* = max\{\mu_G(\mathrm{x})\}$$

In fact, this last operation (maximum) is a defuzzification process, i.e., x^* is the optimal value in the original scale.

In the same way, for the fuzzy minimum function, a sequence can also be structured. Initially, all membership functions are merged by the conjunction "or" (it means the union of all membership functions, operator maximum) and then the optimal value x^* is computed by the operator minimum, as defined by the following sequence

$$\mu_G(\mathrm{x}) = max(C, F)$$
$$\mathrm{x}^* = min\{\mu_G(\mathrm{x})\}$$

In the fuzzy optimization process, it is possible to incorporate weights for each constraint and pseudogoal. These weights can represent linguistic hedges in order to modify a membership function (as a linguistic value). Also, other operators (than maximum and minimum) can be used to define relations among constraints and pseudogoal. Sometimes, composite operators must be used for a better definition of the relations [42].

Numerical Example

This section presents a numerical illustrative example on the use of fuzzy optimization for one-single variable. Let $f(x)$ be an objective function that represents the following linguistic statement "x must be around 4" and the two constraints: C_1 = "x must be equal or greater than 2 and equal or less than 6", and C_2 = "a good value for x is equal or less than 3 and an acceptable value is not much greater". In this example, the former constraint is a crisp function, while the latter constraint is a fuzzy value. Let's consider the example below.

Maximize $\quad f(\mathrm{x}) = 10 - \mathrm{x} - 25/\mathrm{x}^2$

subject to

$$C_1(\mathrm{x}) = \begin{cases} 0 & \text{for x} < 2 \\ 1 & \text{for } 2 \le \text{x} \le 6 \\ 0 & \text{for x} > 6 \end{cases}$$

$$C_2(\mathrm{x}) = \begin{cases} 1 & \text{for x} \le 3 \\ \frac{1}{3} + \frac{2}{\mathrm{x}} & \text{for x} > 3 \end{cases}$$

The initial step is to compute the pseudogoal $F(\mathrm{x})$ using the minimum and maximum values of $f(\mathrm{x})$:

f(x=2) = I = 1.75 (minimum value in the interval [2,6])

f(x=3.68) = S = 4.47 (maximum value in the interval [2,6])

Thus,

$$F(\mathrm{x}) = \frac{f(\mathrm{x}) - \mathrm{I}}{\mathrm{S} - \mathrm{I}} = 3.03 - \frac{\mathrm{x}}{2.72} - \frac{9.19}{\mathrm{x}^2}$$

As the constraints have been defined as membership functions, the next step is compute the membership of the decision function $G(\mathrm{x})$. This computation is performed using the linguistic conjunction "and" because the objective function and the constraints must be satisfied simultaneously. The result is shown in Figure 4, where the minimum operator has been used. The bold curve is the decision membership function.

The final step is the computation of the optimal value of x^* by the maximum relation of $G(\mathrm{x})$. In this case, the maximum value (optimal value) is located in the intersection between the second part of constraint C_2 and the pseudogoal $F(\mathrm{x})$. Equating the two functions, the final value of x^* is equal to 3.2.

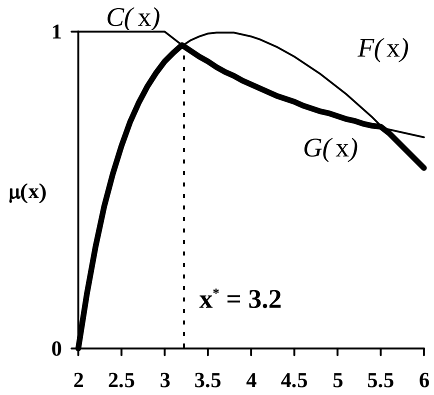


Fig. 4 - Computation of Membership Functions.

C.3 Fuzzy Programming

Fuzzy Linear Programming

Classical linear programming can be defined by an optimization of a linear objective function and linear constraints. Usually, this procedure can be represented by the following statements

$$\text{maximize} \quad f(x) = c^T.x$$

$$\text{subject to} \quad Ax \leq b$$
$$x \geq 0$$

where $c(n\times 1)$, $b(m\times 1)$, $A(m\times n)$ and $m<n$. The inequality constraints form a feasible region.

The fuzzy linear programming has the same structure as the classical linear programming. The difference between these two approaches is that in the classical approach values and operators are crisp, while, in the fuzzy approach values and/or operators may assume fuzzy characteristics. Examples of this fuzzy transformation may be:

- the operator “maximize” cannot specifically be a search for the optimal but only a “improving of quality”,
- the operators ≤ and ≥ can express functions as shown in Figure 2, where for the “belong” crisp region the value of membership is equal to 1, and outside this region, an exponential function defines the membership values, and
- the elements of the vectors b and c and the matrix A can also be a fuzzy definition for a better representation of the real world.

Many contributions have been made in this field, composing the above features [43] and defining new fuzzification and inference processes [44]. Other developments, including duality theory, sensitivity analysis, and integer fuzzy programming can be found in [45].

Fuzzy Dynamic Programming

The idea in classical dynamic programming is to decompose a main problem into several subproblems (one for each variable). Thus, the optimization of each subproblem is divided into a multistage decision process. Here, all operators and values have a crisp meaning. In the same way, a fuzzy dynamic programming can be defined as a fuzzification of all (or part of) these elements. In a well-known fuzzy dynamic programming method, Bellman and Zadeh [41] have proposed to work with fuzzy constraints and fuzzy goals to determine the subgoals of each step of the process, while the transformation function is maintained crisp. An excellent example of the application of fuzzy dynamic programming to power systems is presented in [35].

C.4 Fuzzy Multi-Criteria Analysis

Description of the Problem

During any decision making process, many different factors must be taken into account. These factors can be heuristic or arising from numerical analysis. Usually, the heuristic factors are due to the planner's previous experience and have a non-numerical structure, i.e., they can be better expressed by linguistic values. The problem that planners face in their daily job is how to incorporate these linguistic values into numerical analysis. Commonly, the computational packages do not include the possibility to use non-numerical values. Thus, planners have two possibilities when using this kind of knowledge. One is to put the linguistic numbers in numbers. The other possibility is to forget this knowledge during the numerical analysis and then, after getting the final result, modify it so as to make an adaptation to take into account the planner expertise.

The problem is that both approaches are not good. In the first one, where planner tries to transform linguistic knowledge into numerical values, much information is lost during this process. For example, if the following statement is to be incorporated: "The distribution feeder A is quite loaded." What is a good numerical value for "quite loaded" ? Two possible ways can be taken; that is, the planner uses a number to define it, for example, 0.80 pu, or he/she can use a percentage, say 90%. Here we also lose information in both transformations. In the first one (the worst transformation), if 0.80 pu relates to a 0.85 pu feeder capacity, the statement does not include information about other numbers around 0.80 pu, for example: 0.78, 0.82, and so on. Each of these expresses the same knowledge and have the same result. On the other hand, the number 0.80 alone can not represent "quite loaded feeder", for example, if the feeder capacity is 1.30 pu.

The second representation of the statement, using a percentage or a range, has the same lost of information problem. Let us assume a "small change" in the percentage number; for a long-term decision-making process, it may result in the same final decision. The problem is that it is very hard to quantify 'what is a small change' in a conventional computational tool.

The other possible approach is to modify the final result in order to take into account the planner expertise. This approach has been commonly used in practical analysis; however, planners have had difficulty in explaining why they need to modify a final value, mainly, if this modification can change the final result order given by the decision process.

Classification of Fuzzy Multi-Criteria Analysis

The classification of fuzzy multi-criteria problems is divided in two main types: multi-objective decision-making and multi-attribute decision-making. In general, the difference between these two approaches is located in the decision space. For the former approach, this space is continuous, and the problem is solved by mathematical programming. For the latter approach, the decision space is discrete, and other approaches have been developed [36,46]. The next subsection presents an algorithm to treat this problem.

Presentation of a Multi-Attribute Decision-Making Algorithm

This algorithm is an extension of Dhar's algorithm, proposed in [37]. Some aspects of data structure representation, inclusion of a new matrix composition and a different fuzzy decision-making process are some modifications and extensions proposed in this algorithm. The original algorithm divides the structure of the problem into alternatives, scenarios and criteria, and its matrix representation. Several facilities are included in the user-interface for an easy accomplishment of the tasks.

The steps of the proposed algorithm are presented as follows:

Step 1: Choose the alternatives to solve the problems and the criteria that will be used in the decision-making process.
Step 2: Create scenarios with fuzzy weights for each criterion and give the conjunctions to compose them.
Step 3: Create a matrix by the combination between scenarios and alternatives for each decision criterion. These matrices must contain information about the relation between each scenario and each alternative in the light of each criterion.
Step 4: Create the fuzzy conditional statements to represent possible data-base knowledge.
Step 5: Obtain, for each matrix of Step 3, the fuzzy set Z_i that is formed by the input weights, according to

$$Z_i = \mu_i(x_{j,k}) / p_{j,k}$$

where *i*, *j* and *k* represent criterion, alternative and scenario, respectively; and $p_{j,k}$ is the weight assigned to the alternative *j* for a scenario *k* in a given criterion *i*.
Step 6: Obtain the fuzzy set L_i formed by the weights $p_{j,k}$ that are assigned to the pertinence matrix which, in turn, is given by the ratio between each weight and the largest value among all weights of the same matrix. The following equations express these value, where Λ represents the largest weight of the matrix,

$$L_i = \mu_\Lambda(x_{j,k}) / p_{j,k}$$

$$\mu_\Lambda(j,k) = p_{j,k} / \Lambda$$

Step 7: Obtain from Z_i and L_i a matrix C_i that is expressed by the equations,

$$C_i = \mu_{C_i}(x_{j,k}) / p_{j,k}$$

$$\mu_{C_i}(x_{j,k}) = \min(\mu_i(x_{j,k}), \mu_\Lambda(x_{j,k}))$$

Step 8: Use *max*, *min*, and algebraic sum operators to compose the fuzzy decision set, according to Step 2.
Step 9: Present the final decision set for each criterion, and the total result.

The Steps 5 to 7 has been proposed by Dhar in his original algorithm. More information about the algorithm to build the fuzzy conditional statements to represent data sets can be found in [38,39].

Illustrative Example

The expansion strategy of a generation system is to be analyzed, at long term, for a given region. The generation options are hydroelectric plants (*H*) and nuclear-type thermoelectric plants (*N*), natural gas (*NG*), coal (*C*) and oil fuel (*OF*). This expansion policy is also intended to be associated to investments in electrical power conservation programs trying to establish, within some scenes, an option scale of generation and conservation measures.

The characterization of each plant, under quantitative and qualitative stand point, is shown in Table 1. Some data have been obtained from Brazilian Power System (Eletrobrás) Internal Reports. These values are divided into two groups: numerical values and linguistic values. For the generating system there are construction and generation costs; for the electric power there are the "demand reduction cost" and the "saved energy cost", in (US$/kW) and (US$/kWh), respectively. Tables 2 and 3 illustrate these costs for the

Table 1 - Quantitative and Qualitative Characteristics of the Generation Systems

	Construct. Cost (US$/kW)	O&M Cost (US$/km /year)	Unity Generation Cost (US$/kWh)	Environmental Costs	Generation Reliability	Ease of Implementati on
H	1500	7	0.032	Small	Very High	Small
N	1660	44	0.059	Very High	High	High
NG	1100	22	0.051	Small	Regular	Regular
C	1400	28	0.045	Regular	Regular	Regular
OF	1200	12	0.073	Regular	Regular	Regular

Table 2 - Electrical power conservation costs for industrial sector

	Demand Reduction Cost (US$/kW)	Saved Energy Cost (US$/kWh)
Motors	200-1600	0.02-0.04
Direct Heating (Furnaces)	200-1200	0.02-0.03
Indirect Heating	200-900	0.01-0.02
Electrochemical Processes	200-600	0.01-0.03
Lighting	200-1300	0.02-0.04

Note: Indirect heating includes boiler and water heating.

Table 3 - Comparison of different lighting options

Kind of Lamp	Power (W)	Average Operating Life (hours)	Saved Energy Cost (US$/kWh)	
			A	B
Incandescent Economical (I1)	54	1000	0.027	0.026
Common Tubular Fluorescent (I2)	20	6000	0.031	0.026
Fluorescent Compact (I3)	13	8000	0.060	0.049

Note: A - considering 3 hours/day operation
B - considering 10/hours/day operation

Table 4 - Comparison of different costs of the electrical

Kind of Program	Saved Energy Cost (US$/kWh)
Motor 1 (M1)	0.01
Motor 2 (M2)	0.02
Motor 3 (M3)	0.04

Table 5 - Electrical Power Conservation Scene in the Planning Horizon

Possible States	Description	Membership Degree
HM	Household Medium	0.3
IM	Industrial Medium	0.4
IH	Industrial High	0.8
CL	Commercial Low	0.2

industrial sector and final uses of electrical power [40].

For motors, main electrical power consumer in the industrial sector, it is possible to work in programs of energy conservation which seek, for example, a better used and adaptation in the industrial process (MOTOR 1), the employment of more efficient motors (MOTOR 2) or even the use of varying speed controllers (MOTOR 3) applied to varying torque motors. Each of these options presents different saving energy costs, as shown in Table 4.

Based on the information that we can obtain from Tables 1 to 4, the several technologies aiming at electrical power conservation can be quantitatively and qualitatively characterized within a planning horizon.

Energy Conservation Scenarios and Characteristic Matrices

By attributing a weight from 0 to 10, for example, or a fuzzy linguistic variable, that represents subjectively the importance of each generation plant and the actions of electrical power conservation in the final uses, scenes can be established and the so-called characteristic matrices can be constructed (Table 5).

The characteristic matrices are constructed for the following analysis criteria:
- construction cost or demand reduction cost
- operation and maintenance (O&M) cost;
- generation cost or saved energy cost;
- environmental costs;
- generation reliability or action reliability;
- ease of implementation; and
- usefulness to the entrepreneur.

As an example of this kind of matrix, Table 6 shows the O&M cost characteristic matrix.

Table 6 - Characteristic Matrix - O&M Cost

Alternatives States	HM	IM	IH	CL
H	8	9	10	7
N	2	2	2	2
NG	3	3	4	3
C	2	2	3	2
OF	5	6	7	5
I1	VH	H	H	VM
I2	H	VM	VM	M
I3	H	VM	VM	M
M1	VL	VM	VM	VL
M2	VL	VM	VM	VL
M3	VL	M	M	VL

Calculation of the Fuzzy Decision Set

By using the proposed methodology, the following decision set D is obtained:

D = { (0.6073/H) , (0.4476/N) , (0.6223/NG) , (0.5867/C) , (0.5257/OF) , (0.5692/I1) , (0.5375/I2) , (0.5432/I3), (0.7165/M1) , (0.7032/M2) , (0.5272/M3) }

Thus, for the conditions stated above, the investment strategy in conservation and generation of electrical power is as follows:

1st option: To improve the use and suitability of the motor in the industrial process.
2nd option: To employ more efficient motors in the motors in the process.
3rd option: Hydroelectric generation
4th option: Natural gas thermoelectric generation
5th option: Coal thermoelectric generation
6th option: Nuclear thermoelectric generation

Consider, as an example, that the electric power conservation potential through employment of the 1st option is 8 (TWh) and with the 2nd option is 5 (TWh). Consider also a prediction in the planning horizon of the electrical power market in the order of 60 (TWh). Then, once the 2 first options of conservation are exhausted, there would be a deficit of 47 (TWh). By using the policy of avoiding this deficit only through generation and by considering hydroelectric generation potentials of 25 (TWh), natural gas thermoelectric generation of 10 (TWh) and coal generation of 20 (TWh) yields the following strategy:

Conservation:
13 (TWh), representing 22% of the power demand
Generation:
47 (TWh), representing 78% of the power demand
Conservation actions:

Use 1st and 2nd options

Generation actions:

53% for hydroelectric generation, 21% for natural gas thermoelectric generation, 26% for coal thermoelectric generation

D Fuzzy Control

D.1 Initial Considerations

The use of Fuzzy Logic for solving control problems has tremendously increased over the last few years. Thus, the teaching of fuzzy control in engineering courses is becoming a necessity. This section presents a computational package for students' self-training on fuzzy control theory. The package contains all required instructions for the users to gain an understanding of fuzzy control principles. Training instructions are presented via a practical example. The main objective of the example is to park a car, approaching from any direction, in a parking lot. A car parking problem has been chosen because it is well-known to undergraduate students. To accomplish this task the students must first develop sets of fuzzy-control rules to define the trajectory of the car. Many windows and numerical routines are available in the program to give support to users during the establishment of such fuzzy-control rules. Processes, such as fuzzification and defuzzification of the variables, are performed by the program without the interference of the user. Illustrative examples of training sessions are described in this section.

D.2 Brief Description of the Software

The basic problem of car motion has evolved into a more specific task: the parking of a car in a lot. This problem formulation can exhibit two characteristics: (a) the reduction of the initial data, and (b) the creation of different control zones. The first characteristic shows up because the final position is always the same; the program user needs to enter only the initial car position (coordinates); the final position coordinates are internal data in the program. It is important to note that this reduction does not limit the problem, because all relative positions between the initial and final positions can be maintained. The second characteristic allows us to define fine and coarse control zones for the car motion. For instance, if the current car position is far away from the final position, a coarse output value can be produced without a major implication in the final result. However, if the car is close to the final position, a more rigorous control must be made. The characteristics of fine or coarse control can be represented by the number of rules in each region (non-zero domain of the membership functions). This is an important learning task for the student.

The main idea behind the proposed lab is to demonstrate to undergraduate students that fuzzy logic control is very useful. The strategy for controlling car parking can be the same for many industrial processes [47]. In this lab, each student is asked to built his/her own set of membership functions and control rules. The inference process is performed by the program directly. The problem is to drive the car backward from a given initial position to a target area. The input variables for this problem are: the position of the car trunk (x, y); the car angle ϕ; and the direction of motion d. The output variable, θ, defines the angle of the car front wheels.

D.3 Brief Description of the Software

A fuzzy inference process executes each rule, i.e., (a) the input variables are transformed into the fuzzy statements, and (b) an output value is computed. The execution of each rule is made using *modus ponens* [10], which means that the premise of the rule produces the degree of membership for the conclusion of the same rule. This membership degree is a function of the fuzzy values of the input variables, and of the conjunctions used among them. Figure 5 shows the Rule Editor Window.

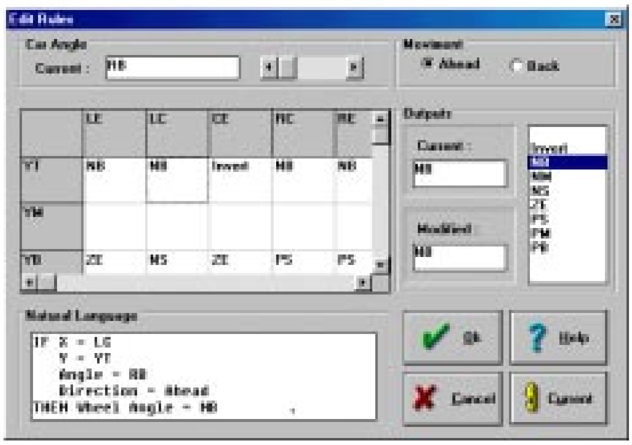


Fig. 5 - Rule Editor Window

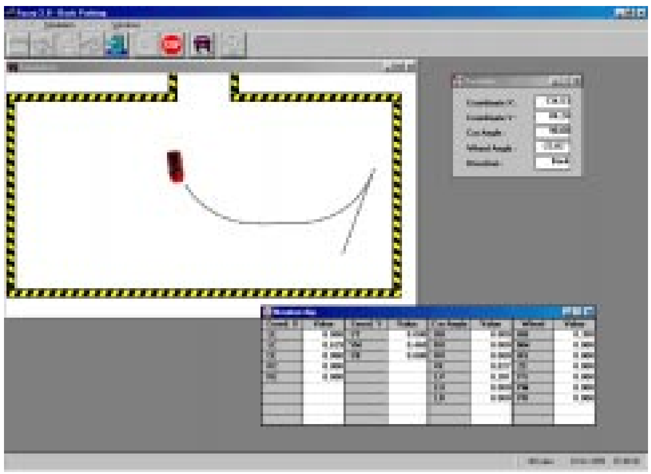


Fig. 6 - Typical Session - Main Window

After the execution of all the rules, the defuzzification process begins. The final actual output value is computed using the center of gravity method [12]. In this method, all areas formed by the consequence of each rule are added and the centroid of the resultant area is computed. The value of the abcissa found is the actual value of the output variable.

D.4 Illustrative Examples

This package allows the students to learn the fundamental principles of fuzzy logic and fuzzy control and, at the same time, to properly choose the number/position of rules for a well-designed controller. A controller with an excessive number of rules (or a lack of them) can compromise the system performance; that is to say, too many correct rules (information or knowledge) can not always be interpreted as an improvement of the system performance as these may worsen it.

A typical situation that may occur in a training session are shown in Figure 6.

References

[1] E. Cox, *The Fuzzy Systems Handbook*, Academic Press, 1994.

[2] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*, Academic Press, 1980.

[3] M. Jamshidi, N. Vadiee, and T. Ross, *Fuzzy Logic and Control*, New Jersey: Prentice Hall, 1993.

[4] M. Jamshidi and T. Ross, *Intelligent and Fuzzy Systems*, New York: John Wiley & Sons, 1994.

[5] P. L. Jones and I. Graham, *Expert Systems: Knowledge, Uncertainty and Decision*, London: Chapman and Hall, 1988.

[6] A. Kandel, *Fuzzy Mathematical Techniques with Applications*, Reading, MA: Addison-Wesley Pub. Co., 1986.

[7] G. J. Klir and T. A. Folger, *Fuzzy Sets, Uncertainty, and Information*, New Jersey: Prentice Hall, 1988.

[8] B. Kosko, *Fuzzy Thinking. The New Science of Fuzzy Logic*, New York: Hyperion, 1993.

[9] F. M. McNeil and E. Thro, *Fuzzy Logic, a Practical Approach*, Academic Press, 1994.

[10] C. Negoita and D.A. Ralescu, *Applications of Fuzzy Sets to Systems Analysis*, New York: John Wiley & Sons, 1975.

[11] C. Negoita, L. Zadeh, and H. Zimmerman, *Fuzzy Sets and Systems*, Amsterdam, the Netherlands: Elsevier Science, 1994.

[12] T.J. Ross, *Fuzzy Logic with Engineering Applications*, New York: McGraw Hill Inc., 1995.

[13] K. J. Schmucker, *Fuzzy Sets, Natural Language Computations, and Risk Analysis*, Rockville, MD.: Computer Science Press, 1984.

[14] T. Terano, K. Asia, and M. Sugeno, *Fuzzy Systems Theory and its Applications*, Academic Press, 1991.

[15] Z. Wang and G. Klir, *Fuzzy Measure Theory*, New York: Plenurn Press, 1992.

[16] R. R. Yager, S. Ovchinnikov, R. M. Tong, and H. T. Nguyen, *Fuzzy Sets and Applications: Selected Papers by L.A. Zadeh*, New York: John Wiley and Sons, 1987.

[17] L. Zadeh K.-S. Fu, K. Tanaka, and M. Shimura, *Fuzzy Sets and Their Application to Cognitive and Decision Processes*, Academic Press, 1975.

[18] H. J. Zimmerman, *Fuzzy Set Theory – and Its Applications*, Boston: Kluwer Academic Publishers, 1991.

[19] H. J. Zimmerman, *Fuzzy Sets, Decision Making, and Expert Systems*, Boston: Kluwer Academic Publishers, 1987.

[20] L. A. Zadeh, Fuzzy sets. *Inf. Control* **8**, 1965, pp. 338-353.

[21] A. K. David and R.D. Zhao, "An Expert System with Fuzzy Sets for Optimal Planning", *IEEE Trans. on Power Systems*, Vol. 4, No. 3, Aug. 1989.

[22] A.K. David, "Expert System with Fuzzy Sets for Optimal Long-Range Operation Planning", *IEEE Trans. on Power Systems*, Vol. 6, No.1, Feb. 1991.

[23] V. Miranda and M.A. Matos, "Distribution System Planning with Fuzzy Models and Techniques", *Proc. of CIRED*, Brighton, Aug. 1989.

[24] G. Lambert Torres, J. Haddad, C.I.A. Almeida & G.M. Jannuzzi, "A Fuzzy Logic Decision-Making Approach for Industrial Power System Planning", *Proc. of IEEE-SMC*, Vancouver, 1995.

[25] K. Tomsovic, G. Skugge, D. Sjelvgren & J. Bulbenko, "Seasonal Planning for Best Fuel Mix under Multiple Objectives using Fuzzy Mathematics", *Proc. of ESAP*, Melbourne, Jan. 1993.

[26] V. Miranda, M. Matos & J.T. Saraiva, "Fuzzy Load Flow - New Algorithms Incorporating Uncertain Generation and Load Representation", *Proc. of 10th PSCC*, Graz, Aug. 1990.

[27] C.E. Lin, C.J. Huang, C.L. Huang & S.Y. Lee, "An Optimal Generator Maintenance Approach using Fuzzy Dynamic Programming", *IEEE/PES Summer Meeting*, Paper 92SM401-0PWRS, Seattle, July 1992.

[28] C.C. Su and Y.Y. Hsu, "Fuzzy Dynamic Programming: An Application to Unit Commitment", *IEEE Trans. on Power Systems*, Vol. 6, No.3, Aug. 1991.

[29] T. Mackaw, K. Yamuda & R. Yokohama, "Fuzzy Coordination of Economy and Quality in Power System Generation Control", *Proc. of the Third Int. Conf. On Power System Monitoring and Control*, London, June 1991.

[30] T. Mackaw, K. Yamuda, R. Yokohama, H. Obtsuki & Y. Mitzukami, "Fuzzy Coordination of Multi-Objective Refleting Operator's Intention for Dynamical Generator Rescheduling", *Proc. of ESAP*, Toquio/Kobe, April 1991.

[31] I.J. Ramirez-Rosado and J.A. Domínguez-Navarro, "Distribution Planning of Electric Energy ysing Fuzzy Models", *Int. Journal of Power and Energy Systems*, Vol. 16, No.2, 1996.

[32] V. Miranda and J.T. Saraiva, "Fuzzy Modeling of Power System Optimal Power Flow", *IEEE Trans. on Power Systems*, Vol.7 No.2, May 1992.

[33] Y. Terasawa and S. Iwamoto, "Optimal Power Flow using Fuzzy Mathematical Programming", *Electrical Engineering in Japan*, Vol. 108, No.3, 1988.

[34] K.H. Abdul-Rahman and S.M. Shahidehpour, "Application of Fuzzy Sets to Optimal and Reactive Power Planning with Security Constraints", *Proc. of PICA*, Arizona, May 1993.

[35] J.A. Momoh, X.W. Ma & K. Tomsovic, "Overview and Literature Survey of Fuzzy Set Theory in Power Systems", *IEEE Trans. On Power Systems*, Vol. 10, No.3, Aug. 1995.
[36] D. Muñoz-Rodriguez and K.W. Cattermole, "Multiple Criteria for Hand-off in Cellular Mobile Radio"¸ *IEE Proc.*, Vol. 134, 1987.
[37] S.B. Dhar, "Power System Long-Range Decision Analysis under Fuzzy Environment". *IEEE Trans. on Power Apparatus*, Vol. PAS-98, No. 2, 1979.
[38] G. Lambert-Torres, R. Kosztaluk & D. Mukhedkar, "Short-Term Feeder Load Forecasting using Fuzzy Environment", *Systems Science Journal*, Vol. 17, No. 2, pp. 51-62, 1991.
[39] G. Lambert-Torres, L.E. Borges da Silva, B. Valiquette, H. Greiss & D. Mukhedkar, "A Fuzzy Knowledge-Based System for Bus Load Forecasting", In: Fuzzy Logic Technology and Applications, edited by Robert J. Marks II, *IEEE Press*, pp. 221-228, New York, 1994.
[40] J. Haddad, G. Lambert-Torres, C.I.A. Costa & G.M. Jannuzzi, "Long-Term Power System Planning Using a Fuzzy Treatment", *Proc. of IFSA*, São Paulo, July 1995.
[41] R. Bellman and L.A. Zadeh, "Decision-making in a Fuzzy Environment", *Management Science*, Vol. 17, 1970.
[42] K. Asai, H. Tanaka and T. Okuda, "Decision Making and Its Goal in a Fuzzy Environment", In: *Fuzzy Sets and Their Applications to Cognitive and Decision Processes*, New York, 1975.
[43] H.J. Zimmermann, "Fuzzy Mathematical Programming", *Computational Operational Research*, Vol. 10, 1983.
[44] B. Werners, "Aggregation Models in Mathematical Programming", In: Mathematical Models for Decision Support, Berlin, 1988.
[45] H.J. Zimmermann, "Using Fuzzy Sets in Operational Research", *EJOR*, Vol. 13, 1983.
[46] C.L. Hwang and K. Moon, *Multiple Attribute Decision Making*, Berlin, 1981.
[47] B. Kosko - *Neural Networks and Fuzzy Systems*, Prentice Hall, Eaglewood Cliff, New Jersey, 1991.

Chapter 3 Control Applications

Abstract- The basic concepts of fuzzy systems, particularly from the point of view of fuzzy logic based controllers, are described in this Chapter. State of the art of fuzzy control for power systems is outlined and supported by a bibliography of the literature in this area. Application of the fuzzy logic controller as a power system stabilizer is illustrated by two examples - one fairly straightforward application of fuzzy logic and one self-learning algorithm.

A Introduction to Power System Control

A.1 Motivation

A reliable, continuous supply of electric energy is essential for the functioning of today's complex societies. Due to a combination of increasing energy consumption and impediments of various kinds concerning the extension of existing electric transmission networks, these power systems are operated closer and closer to their limits.

Deregulatory efforts will tighten the economical constraints under which utilities have to operate their own network or allow or prevent competitors from using it. This in turn will require more precise power flow control which is made possible by phase angle controllers being developed using new power electronic equipment. However, it is to be expected that these highly non-linear components will introduce harmonics and require non-linear control in order to prevent system destabilization.

This situation requires a significantly less conservative power system operation regime which, in turn, is possible only by monitoring and controlling the system state in much more detail than was necessary previously.

A.2 Power System Control Tasks

In electric power systems, [1], one can distinguish three different control levels:

Generating Unit Controls which consist of prime mover control and excitation control with automatic voltage control (AVR) and power system stabilization (PSS). The first controls generator speed deviation and energy supply system variable like boiler pressure or water flow. Excitation control aims at maintaining the generator terminal voltage and reactive power output within its machine-dependent limits.

System Generation Control which determines active power output such that the overall system generation meets the system load. It further controls the frequency and the tie line flows between different power system areas.

Finally *Transmission Control* monitors power and voltage control devices like tap-changing transformers, synchronous condensers and static VAR compensators.

In reality all controls affect both components and systems. For example the AVR is known to introduce local mode oscillations as well as inter-area oscillations which in turn are counteracted by a well-tuned PSS.

From the viewpoint of system automation, Generating Unit Control is a complete closed-loop system and in the last decade a lot of effort has been dedicated to improve the performance of the controllers. The main problem for example with excitation control is that the control law is based on a linearized machine model and the control parameters are tuned to some nominal operating conditions. In case of a large disturbance, the system conditions will change in a highly non-linear manner and the controller parameters are no longer valid. In this case the controller may even add a destabilizing effect to the disturbance by for example adding negative damping.

These problems provide an important motivation to explore novel control techniques like fuzzy systems and their potential in the area of prediction, approximation, classification and control.

Power system control consists of 4 steps:

1- System parametric or state-space modeling based on physical components or assumed properties
2- System parameter identification based on component data and measurements.
3- System observation of inputs and outputs by filtering, prediction, state estimation etc.
4- Design of an open-loop or closed-loop system control law such that the operating conditions are met.

In the case of electric power systems and electric machines, individual components are modeled in terms of resistors, inductors, capacitors, machine inertia etc. Their interaction is modeled according to the laws of electro-magnetic circuits and fields. The resulting set of differential equations then defines a state-space model whose parameters, for example the machine reactances have to be identified under steady state and transient conditions. Voltage signals on the other hand are modeled as a trignometric sum of sin and cos functions without taking the underlying physical model into account. In this case the free parameters to be identified are the signal amplitudes.

System identification may be defined as the process of determining the parameters of the dynamic system model using observed input and output data. Dynamic load modeling attempts to model individual as well as composite loads of the system. Identification of machine parameters is another identification task.

System observation in power systems concerns off-and on-line monitoring of directly or indirectly observable system variables. Load forecasting, for example is an off-line monitoring task, power quality monitoring an on-line monitoring task. *State estimation* calculates the most likely values for power system parameters like bus voltage and line flow by giving a least-squares estimate of a set of redundant measurements.

Closed-loop control attempts to counter-balance undesirable effects like undamped frequency oscillations or voltage deviations in a closed-loop feedback environment. Excitation control, automatic voltage regulation and power system stabilization fall in this category.

Assume that the plant has been modeled by the following single-input, single-output noise-free continuos system and the parameters of the non-linear plant model f and observer model h have been identified.

$$d\mathbf{x}(t)/dt = f(\mathbf{x}, u, t)$$
$$y = h(\mathbf{x}, u, t)$$

where

$\mathbf{x} \in \Re^m$ is the state vector of the modeled plant
$u \in \Re$ is the control input of the system
$y \in \Re$ is the observed output of the system
f is a non-linear state-space model of the plant
h is the non-linear observer model

Figure 1 shows an example of a simple continuous closed-loop control system where r is a reference signal, $\phi(e,t)$ denotes the controller and e = r-y is the feedback error to be minimized.

In the case of a power system stabilizer the reference signal is the reference voltage V_{ref} and the terminal voltage V_t, the plant output is the angular speed deviation $\Delta\omega$, and the control signal is the stabilizing voltage v_S. The plant model comprises excitation system, AVR and generator model.

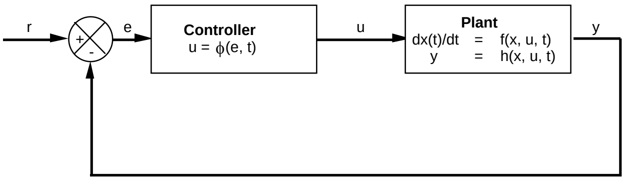


Figure 1: Closed-loop control system

So far the continuous system model has been considered. If the system in Figure 1 is discretized taking into account k = 1...n time-steps t=kT, the control, output and error signals will become higher -dimensional vectors, as for example

$$\mathbf{u} = [u(k), u(k-1), ... u(k-m)]^T$$
$$\mathbf{e} = [e(k), e(k-1), ...e(k-n)]^T$$
$$y(k) = h_d(\mathbf{x}(k), \mathbf{x}(k-1), ..., u(k), u(k-1) ...)$$

In the case of feedback error minimization, task 4 of the controller design consists in finding a function

$$\phi: \Re^n \rightarrow \Re \text{ such that } \phi(\mathbf{e}) = u(k+1) \text{ and } |e(k+1)| = \min.$$

In the case of linear controllable and observable systems, a controller ϕ can be found through inversion of the system transfer function and pole placement. In the case of non-linear systems there is no general closed form of ϕ. How fuzzy systems can provide an approximation of the controller ϕ is explored in the following sections.

B Fuzzy Systems

B.1 Review of Basic Concepts

Some of the basic definitions of fuzzy systems as outlined in [2] are reviewed first. Let $U \subseteq \Re^n$ denote the universe of discourse, with fuzzy sets $A \subseteq U$ and fuzzy membership functions μ_A: U -> [0,1] which may be labeled by a linguistic term like cold, warm, positive or negative.

Fuzzy membership functions are characterized by their shape and their localization in space. For example, if the universe of discourse is the range of temperatures from 0 F to 120 F, the membership function describing the fuzzy set cold, warm and hot may be centered at a_1 = 40 F, a_2 = 70 F and a_3 = 100 F, have a triangular shape of slope and a maximal width σ of 40 F as shown in Figure 2.

Instead of defining center, shape and width of the

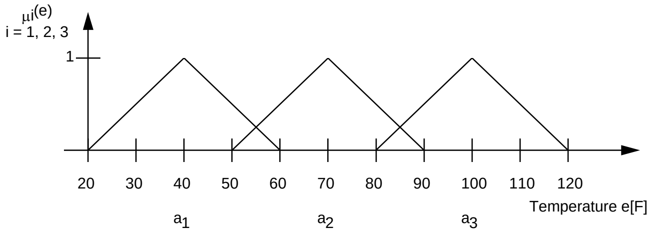


Figure 2: Membership Function for Fuzzy Temperature Sets

membership function by empirical rules, one can choose a more systematic approach using data analysis. For example, in the case of load forecasting sampling of the load data might indicate that the load exhibits 3 different behaviors correlated with the temperature. A clustering algorithm might have identified 3 typical temperatures a_i and the width of the cluster σ_i defines the width of the membership function μ_i, i=1, 2, 3.

In addition, one can choose a *Gaussian function*, which is continuously differentiable, instead of the triangular (or the sometimes used trapezoidal shape) without altering the quality of the results significantly. (Note, however that the support of the triangular function is finite whereas the support of the Gaussian is the whole universe of discourse.)

$$\mu_{A_i} = s_i * \exp(-0.5[(e - a_i) / \sigma_i]^2$$

There are other continuously differentiable membership functions available with finite support available, for ex. B-splines. Whether one defines the membership function empirically or systematically, one always has some degrees of freedom, that is

N: number of fuzzy sets and membership functions
a_i: the center of the μ_i
σ_i: the width of μ_i or s_i: the slope of μ_i under the constraint that $\mu_i(e) \in [0,1]$.

In analogy to crisp sets one can define *union*, *intersection* and *complement* of two fuzzy sets A and B by defining the membership functions corresponding to union, intersection and complement.

The most widely used definitions are the *min-max* operations

Intersection: $\mu_{A \cap B}(e) = \min(\mu_A(e), \mu_B(e))$
Union: $\mu_{A \cup B}(e) = \max(\mu_A(e), \mu_B(e))$
Complement: $\mu_{U \backslash A}(e) = 1 - \mu_A(e)$

These definitions of union and intersection are not unique and depending on the choices one can define different fuzzy systems with different rule inference mechanisms. Another popular choice of intersection and union is the *algebraic product and sum* operations:

Intersection: $\mu_{A \cap B}(e) = \mu_A(e) \cdot \mu_B(e)$
Union: $\mu_{A \cup B}(e) = \mu_A(e) + \mu_B(e) - \mu_A(e) \cdot \mu_B(e)$
Complement: $\mu_{U \backslash A}(e) = 1 - \mu_A(e)$

Here these two types of intersection definitions are referred to as *minmax T-norm* T and p*roduct T-norm* T.

Minmax: $T(\mu_A, \mu_B)(e) = \min(\mu_A(e), \mu_B(e))$
Product: $T(\mu_A, \mu_B)(e) = \mu_A(e) \cdot \mu_B(e)$

The corresponding unions are called *minmax T-conorm* and *product T-conorm*. Reference [3] gives examples of other T-norms and their implications on rules and inferences.

As in the case of the choice of membership functions these T-norms have different advantages. The minmax T-norm is closer to the set-theoretic approach and generalizes the concept of crisp union and crisp intersection in a rather intuitive manner. The product T-norm implements the logic AND and OR calculation for boolean values 0 and 1.

However, even for differentiable membership functions, the membership function of the set intersections generated by the minmax T-norm is, in general, not differentiable whereas the membership functions of intersections generated by the product T-norm remain differentiable.

The definition of intersection and union leads to the definition of fuzzy compositions, propositions and fuzzy implication interpreted according to the chosen T-norm. See also Chapter 3. The most important definitions are:

Fuzzy proposition
e_1 is A AND e_1 is B <==> e_1 is $A \cap B$

Fuzzy composition
e_1 is A_1 AND e_2 is A_2 <==> (e_1, e_2) are $A_1 \times A_2$
with membership function
$\mu_{A_1 \times A_2}(e_1, e_2) = T(\mu_{A_1}(e_1), \mu_{A_2}(e_2))$

Fuzzy implication:
$\mu_{A \to B}(e,u) = T((\mu_A(e), \mu_B(u))$

Fuzzy rule inference:
Let E_1 and E_2 be the fuzzy sets presenting the input **e** = (e_1, e_2) then the rule IF e_1 is A_1 AND e_2 is A_2 THEN u is B defines a fuzzy set C with the membership function

$$\mu_C(u) = \sup_{e \in U \times U}\{T((\mu_{A_1 \times A_2 \to B}(\mathbf{e}, u), \mu_{E_1 \times E_2}(\mathbf{e}))\}$$

Firing of m fuzzy rules:
Since each rule i will result in a fuzzy set C_i, the firing of m rules for **e** results in a *union* of fuzzy sets
$C = C_1 \cup ... \cup C_m$
with membership function
$\mu_C(u) = \mu_{C_1 \cup ... \cup C_m}(u)$

In the context of power system control, control input and feedback error are given as real "crisp" numbers. In order to apply fuzzy concepts the feedback error **e** has to be presented by a fuzzy set. This set is modified according to the fuzzy rules into a fuzzy set of controls C which are translated back into a crisp control input $u \in \Re$. This task is performed by a fuzzy system defined according to [2] and illustrated in Figure 3.

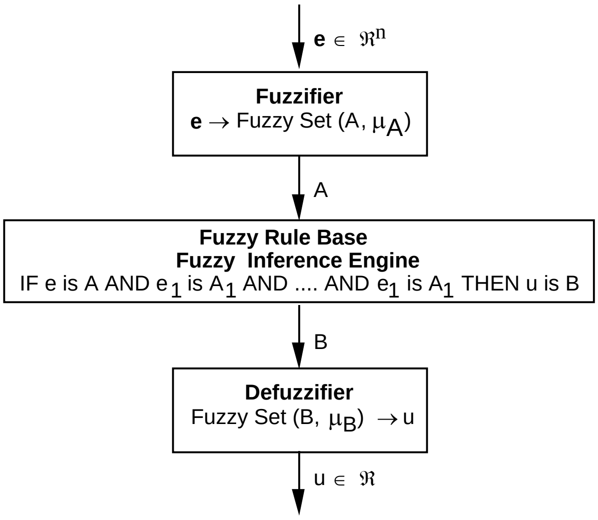


Figure 3: Structure of a Fuzzy System

A *fuzzy system* is a mapping

$$F: U^n \subseteq \Re^n \to \Re, \; F(\mathbf{e}) = u$$

defined on a universe of discourse U^n with fuzzy sets A, membership functions μ_A, and a T-norm T. It is further defined by a fuzzifier, a fuzzy rule base, a fuzzy inference engine and a defuzifier.

A *Fuzzifier* maps $\mathbf{e} \in U^n \subseteq \Re^n$ to a fuzzy set E. A fuzzifier is called the *singleton fuzzifier* if the membership function μ_E of E is defined as

$$\mu_E(\mathbf{x}) = 0, \; \mathbf{x} \neq \mathbf{e}, \; \mu_E(\mathbf{e}) = 1$$

Another fuzzifier is μ_E defined as a Gaussian function with center **e**. For the singleton fuzzifier and crisp inputs $\mathbf{e} = (e_1, e_2)$ the fuzzy inference rule can be simplified as

$$\begin{aligned} \mu_C(u) &= \\ &\sup_{\mathbf{e} \in U \times U} \{T((\mu_{A_1 \times A_2 \to B}(\mathbf{e}, u), \mu_{E_1 \times E_2}(\mathbf{e}))\} \\ &= \mu_{A_1 \times A_2 \to B}(\mathbf{e}, u) \\ &= T(T(\mu_{A_1}(e_1), \mu_{A_2}(e_2)), \mu_B(u)) \end{aligned}$$

Keep in mind that u is a variable while e_1 and e_2 are numbers and thus $\mu_B(u)$ is a function whereas $\mu_{A_1}(e_1)$ and $\mu_{A_2}(e_2)$ are numbers modifying the shape of $\mu_B(u)$. For the product T-norm $\mu_{A_1}(e_1)$ and $\mu_{A_2}(e_2)$ will scale the function $\mu_B(u)$ by reducing the slope.

$$\mu_C(u) = \mu_{A_1}(e_1)\, \mu_{A_2}(e_2)\, \mu_B(u) \qquad (1)$$

If it is assumed that $\mu_{A_1}(e_1) < \mu_{A_2}(e_2)$, then the function $\mu_B(u)$ will be clipped to $\mu_{A_1}(e_1)$ for the minmax T-norm.

$$\mu_C(u) = \min(\mu_{A_1}(e_1), \mu_{A_2}(e_2), \mu_B(u))$$

Thus

$$\begin{aligned} \mu_C(u) &= \mu_B(u) && \text{if } \mu_B(u) < \mu_{A_1}(e_1) \\ \mu_C(u) &= \mu_{A_1}(e_1) && \text{if } \mu_B(u) \geq \mu_{A_1}(e_1) \end{aligned}$$

As outlined above, the application of m rules to the fuzzy input vector **e** results in the union of fuzzy sets $C = C_1 \cup \ldots \cup C_m$. A *defuzzifier* maps the fuzzy output sets of the inference engine onto a crisp number. Once again several choices are available. The *maximum defuzzifier* of the fuzzy set $C = C_1 \cup \ldots \cup C_m$ is defined as

$$u = \operatorname{argsup}_{u \in U} (\mu_{C_1 \cup C_2 \ldots \cup C_m})$$

However depending on the form of the membership function, the maximum may be reached at several points and therefore the determination of u may become ambiguous.
In control the following two defuzzifiers are the most common. For a finite universe of discourse with M elements $U = \{u_1, \ldots, u_M\}$ the *center-of-gravity defuzzifier* is defined as

$$u = \frac{\sum_{i=1}^{M} u_i \mu_C(u_i)}{\sum_{i=1}^{M} \mu_C(u_i)}$$

For a continuos universe of discourse the sum has to be replaced by an integral and the computation of this integral is considerably time-consuming.

A computationally more economic choice of a defuzzifier is the so-called *center-of-average* defuzzifier. Let m be the number of rules and let b_i denote the center of the fuzzy sets B_i where μ_{B_i} reaches its maximum (usually 1), i = 1...m. For one-level inference (each rule is at most fired once) these centers depend only on the initial shape of μ_{B_i} and they can be computed off-line in advance. Then the defuzzifier is defined as

$$u = \frac{\sum_{i=1}^{m} b_i \mu_{C_i}(b_i)}{\sum_{i=1}^{M} \mu_{C_i}(b_i)} \tag{2}$$

B.2 Fuzzy Basis Functions

In [2] it is shown that a fuzzy system F with singleton fuzzifier, product-T-norm and center-of-average defuzzifier can be written in a closed form as

$$u = F(e_1, e_2, \ldots, e_n) = \sum_{j=1}^{m} b_j \, \phi_j(e_1, e_2, \ldots, e_n)$$

where

$$\phi_j(e_1, \ldots, e_n) = \frac{\prod_{i=1}^{n} \mu_{A_{ij}}(e_i)}{\sum_{j=1}^{m} \prod_{i=1}^{n} \mu_{A_{ij}}(e_i)}$$

This result can be derived by generalizing the product inference rule (1) for m rules and by substituting (1) into equation (2). Further, it is assumed that $\mu_{B_i}(b_i) = 1$.

As above, the index j denotes the number of rules used for evaluating the fuzzy input **e** = (e_1, ..., e_n) and b_j indicate the centers of gravity of the fuzzy sets B_j where B_j determines the control output u as an implication of the rules fired **e.**

For Gaussian membership functions

$$\mu_{A_{ij}}(e_i) = s_{ij} * \exp(-0.5[(e_i - a_{ij}) / \sigma_i]^2,$$

the fuzzy system is a universal approximator of any continuos function. This result shows that this specific class of fuzzy logic systems belongs to the class of adaptive controllers and, therefore, that Fuzzy Systems in general can be viewed as a generalization of adaptive controllers.

It is important to note that in this case the only important information one needs to know about the membership function of B_j is its center b_j whereas the Gaussian membership functions defining the fuzzy set A_{ij} for component e_i and rule j have to be defined in detail.

Above a procedure has been outlined on how to determine the centers a_{ij} and width σ_{ij} of the membership functions of A_{ij} with clustering algorithms and without empirical description. In analogy to neural networks, this process is also referred to as *unsupervised or self-organized or self-adaptive training* of the fuzzy system.

An algorithm which calculates the parameters a_{ij}, b_j and σ_i based on the sampled system data using a least-squares minimization algorithm based on the Gram-Schmidt-orthogonalization procedure is described in [2]. The degree m of the system, i.e. the number of rules has to be chosen as a free parameter. In terms of fuzzy systems this algorithm can be viewed as a *supervised or adaptive training* of the fuzzy rule base.

Fuzzy systems given in closed form, whose membership functions and inference rules are established in such a systematic unempirical manner, have the advantage that stability analysis can be performed and tasks like optimal control can be addressed. This is outlined in more detail in chapter 6.

On the other hand, the empirical establishment of the membership functions and the rule base allow one to incorporate available human knowledge and heuristics. For the case of a simple non-linear dynamic system, the “ball-and-beam" system, it is shown in [2] that for undersampled plants additional empirical rules can improve the controller's performance. However, a small set of purely empirical rules may lead to unstable plant behavior.

Table 1: Overview of Fuzzy Systems Applications to Power System Control

Application	References	Fuzzy Approach	Membership Functions	Comments
PSS	[4,5]	Self-adaptive	B-Spline	1-machine, 2-line-inf. Bus; Lab experiment, micro machine system DSP controller
PSS	[6,33]	Rule-based	Triangular	1-machine, 2-line-inf. Bus; Lab experiment, micro machine system DSP controller
PSS	[7-10,12-14]	Self-adaptive	Trapezoidal	Simulation on Analog Power System Simulator (12 machine max); Prototype field test on 2 hydro units, Frequency response study, capacitor bank switching
PSS	[15]	Self-adaptive	Gaussian	Computer simulation, utility power system
AVR & PSS	[16]	Rule-based	Trapezoidal	Computer simulation, 3-machine test system
PSS	[17-18]	Rule-based	Triangular	Computer simulation, 2-machine, 4-line-inf. bus
PSS	[19]	Self-adaptive	Triangular	Computer simulation, 1-machine, 2-line-inf. bus
PSS	[20]	Rule-based	Triangular	Computer simulation, 3-machine, 7-line-inf. bus
FACTS	[11,21]	Self-adaptive	Trapezoidal	Computer simulation, 5-machine, 13-line-inf. Bus, capacitor bank switching, thyristor controlled braking resistor, static VAR compensator
Induction Motor	[22]	Rule-based	Trapezoidal	Lab inverter/3 hp induction motor, PC-based micro-controller
Variable speed drive	[23,24]	Rule-based	Triangular	Lab inverter/reduced ratings, induction motor, DSP-based micro-controller
PWM Inverter	[25]	Rule-based	Triangular	Lab prototype of wind energy conversion system

C. State of the Art of Fuzzy Control for Power Systems

In the case of power systems, control measurement data can be obtained for the discretized plant output **y**, the reference signal **r** and the control input **u.** In analogy to neural networks, let this data be referred to as the training data.

A short overview of the studies of fuzzy systems of type **u** = F(**e**) as controllers ϕ(e) in the area of power systems or generation control is now given. The majority of fuzzy controllers can be found in the area of excitation control, especially power system stabilizers (PSS). An upcoming important area is control of FACTS devices like thyristors and GTOs. Table 1 gives an snapshot of the state-of-the-art. Given the considerable number of publications Table 1 can not claim to give a complete overview. Instead this short summary intends to provide the reader with some information on a typical approaches and project states.

Papers published by the same team of researchers on the same topic have been regrouped to projects. The comments concerning studied system size and test usually report on information given in the latest publication. Although the majority of projects perform feasibility studies using computer simulation only, several authors study the implementation of the fuzzy controller on a PC or DSP in order to control actual small generators or motors in an experimental laboratory environment. In most cases the membership functions are established based on data samples. Those approaches listed as rule-based attempt to justify the fuzzy sets in terms of linguistic descriptions like "if angle is small then deviation should be small".The comparison of fuzzy controllers and conventional controllers stresses advantages of fuzzy controllers as being "generic" parametric models instead of circuit based state space models. The self-adaptive controllers can be easily tuned to different operating conditions and all projects report better tracking capabilities of the fuzzy controllers when compared to conventional controllers. However, the sensitivity issues concerning the range of validity of the tuning and the detection of changes of operating conditions still need to be investigated for conventional as well as for fuzzy controllers. This is especially important for power system control where topology, load and generation can change stochastically and discontinuously.

D. Application of a Fuzzy Logic Controller as a Power System Stabilizer

The design process of the fuzzy logic controller (FLC) has five steps:

- selection of the fuzzy control variables
- membership function definition
- rule creation
- inference engine, and
- defuzzification strategies

To design the FLC, variables which can represent the dynamic performance of the plant to be controlled should be chosen as the inputs to the controller. In addition to the proper input signals, signal gains and fuzzy subsets should be defined. It is common to use the output error (e) and the rate or derivative of the output ($e^{'}$) as controller inputs.

In the case of the fuzzy logic based power system stabilizer (FPSS), the generator speed deviation ($\Delta\omega$) and its derivative ($\Delta\dot{\omega}$), the acceleration, are considered as the inputs of the FPSS. After sampling, two appropriate gains, SG and AG are applied to speed deviation and acceleration, respectively, and then fed to the FPSS. The output of the controller is also scaled by an output gain, UG, and added to the AVR input signal.

The measured input variables are converted into suitable linguistic variables. In this case, seven fuzzy subsets, NB (Negative Big), NM (Negative Medium), NS (Negative Small), Z (Zero), PS (Positive Small), PM (Positive Medium) and PB (Positive Big) have been chosen. Membership functions for the input variables used here are shown in Figure 4. These membership functions are symmetrical and each one overlaps with the adjacent functions by 50%.

Table 2: FPSS Control Rule Table

	$\Delta\omega$							
		NB	NM	NS	Z	PS	PM	PB
	NB	NB	NB	NB	NM	NM	NS	Z
	NM	NB	NB	NM	NM	NS	Z	PS
	NS	NB	NM	NS	NS	Z	PS	PM
$\Delta\dot{\omega}$	Z	NM	NM	NS	Z	PS	PM	PM
	PS	NM	NS	Z	PS	PS	PM	PB
	PM	NS	Z	PS	PM	PM	PB	PB
	PB	Z	PS	PM	PM	PB	PB	PB

In practice, the membership functions are normalized in the interval [-L, L] which is symmetrical around zero. Thus, control signal amplitudes (fuzzy variables) are expressed in terms of controller parameters (gains). These parameters can be defined as:

$$K_j = 2L/X_{range\ j}$$

where $X_{range\ j}$ defines the full range of the control variable X_j. In this study, both inputs of the FPSS have seven subsets. Thus, a fuzzy rule table with forty-nine rules should be constructed. A rule table which is formulated based on the past experience of manual tuning of a conventional PSS (CPSS) is shown in Table 2.

In the next step, the controller output is computed by the inference mechanism. As an example, consider a pair of $\Delta\omega$ and $\Delta\dot{\omega}$ inputs to the controller. In fuzzification stage these inputs are converted to membership grades for each of the seven subsets, e.g. μ_{ω}(PB), μ_{ω}(PM), etc., and $\mu_{\dot{\omega}}$(PB), $\mu_{\dot{\omega}}$(PM), etc. Thus, there are a set of forty nine pairs of membership grades for each of input pair. The smaller element in each pair would be the grade of membership for any of the possible control actions. For example, the FPSS output membership grade for the first rule in Table 2 is given by:

$$\mu_{out}^{1} = \min[\mu_{\omega}(NB), \mu_{\dot{\omega}}(NB)]$$

The output of the FPSS is limited to 0.1 pu and is divided in seven subsets. Also, the output membership functions are chosen as singleton functions as indicated in Table 3.

Table 3 Output Membership Functions

Output subsets	NB	NM	NS	Z	PS	PM	PB
u_{out} pu	-0.10	-0.06	-0.03	0.0	0.03	0.06	0.10

The output of the inference process at this stage is a fuzzy set. In order to take a nonfuzzy (crisp) control action, the fuzzy control action inferred from the fuzzy control algorithm must be defuzzified. Three different defuzzification methods, the Max criterion method, the Mean of Maximum method and the Centre of Gravity method are commonly used [26]. To ensure that all of the fired rules have some contribution in the output control action, the Centre of Gravity method, using the following equation, is employed in this study:

$$u_{pss} = \frac{\Sigma_{i=1}^{Rules}\; u_{out}(Z)\; \mu_{out}^{i}(Z)}{\Sigma_{i=1}^{Rules}\; \mu_{out}^{i}(Z)}$$

where $\mu_{out}^{i}(Z)$ denotes the output membership grade for *i*th rule with the output subset of Z. To achieve the best performance with FPSS, the input and output gains need to be selected properly. For this purpose, the speed deviation and its derivative were measured for a variety of small and large disturbances applied to the system.

The universe of discourse for both inputs of the FPSS is normalized as shown in Figure 4. Therefore, appropriate gains should be chosen such that they map the measured inputs of the FPSS to their suitable linguistic variables. For example, for a small disturbance the measured inputs should be mapped to the "Small" domain, whereas for a large disturbance they should be mapped to the saturated region of the "Large" domain.

It was found for the system under study that for different applied disturbances on the system, the magnitude of $\Delta\dot{\omega}$ was about ten times that of $\Delta\omega$. As the same membership functions are used here for both inputs, the input gain for $\Delta\omega$ should be about ten times the input gain for $\Delta\dot{\omega}$. After fixing the input gains, the output gain should be selected such that the controller is sensitive to the errors in the lower region of the universe of discourse. At the same time, to minimize the time the control stays in the saturated region of the controller, the output gain selected should not be very high.

The performance of the FPSS was studied, both by simulation studies on a seventh order non-linear model and by experimental studies in the laboratory on a physical model of a single machine infinite bus system. For experimental studies the FPSS was implemented on a single board computer, Intel iSBC386/21. In all these studies, the sampling period was set at 10ms. Studies were performed with the FPSS and a fixed parameter CPSS for a variety of disturbances over a wide range of operating conditions. The control output of both stabilizers was limited to 0.1 pu in all studies. The parameters of the CPSS were tuned such that the system response with the CPSS was practically the same at the nominal operating point. Illustrative experimental results for a 4.5% step decrease in voltage reference and return to initial condition at a heavily loaded operating point of P = 1.12 pu, 0.98 pf lag are shown in Figure 5.

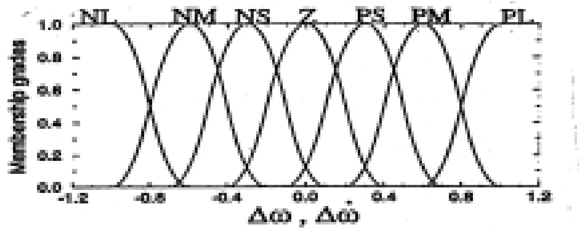


Figure 4. Membership Functions for Δ ω and Δ ω̇

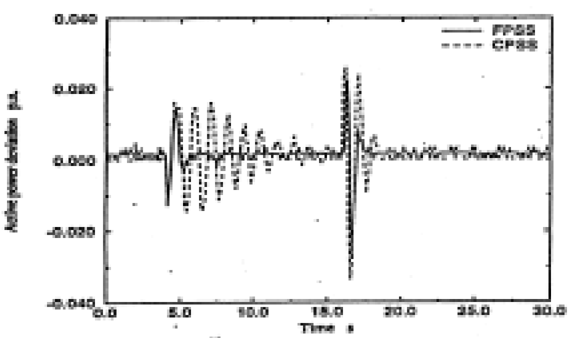


Figure 5. Response to a 4.5% step decrease in voltage reference and return to initial condition, P = 1.12 pu, pf = 0.98 lag

E. A Self-learning Fuzzy Power System Stabilizer

A lot of effort is required in the creation and tuning of the fuzzy rules for an FLC which can be time consuming and non-trivial [26]. A self-learning adaptive network can be used to reduce this effort. A class of adaptive networks which are functionally equivalent to FLC combines the idea of the FLC and adaptive network structures [27]. As a result an FLC network can be constructed automatically by learning from the training examples.

Essentially, an adaptive network is a superset of multi-layer feedforward network with supervised learning capability. The network consists of nodes and directional links through which the nodes are connected. Each node performs a particular function which may vary from node to node. In this network, the links between the nodes only indicate the direction of flow of signals and a part or all of the nodes contain the adaptive parameters. These parameters are specified by the learning algorithm and should be updated to achieve the desired input-output mapping.

An adaptive network based FLC employed as a fuzzy logic PSS (ANFPSS), Figure 6, has two inputs, the generator speed deviation and its derivative, and one control output [28]. The node functions in each layer are:

- layer 1 performs a membership function
- layer 2 represents the firing strength of each rule
- layer 3 calculates the normalized firing strength of each rule
- layer 4 output is the weighted consequent part of ` the rule table
- layer 5 computes the overall output as the summation of all incoming nodes.

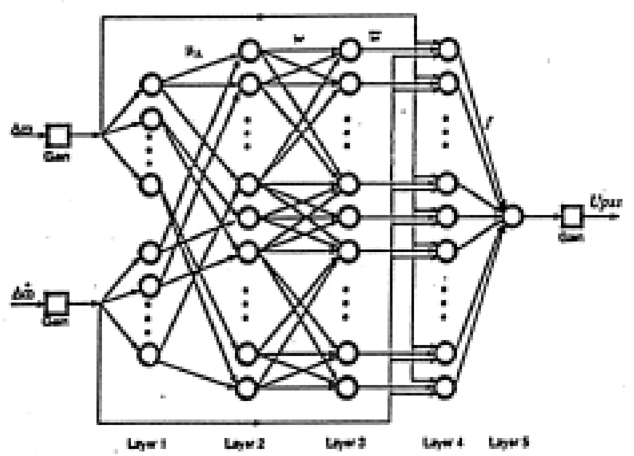


Figure 6. Architecture of ANF PSS

This adaptive network is functionally equivalent to a fuzzy logic PSS. Because the adaptive network has the property of learning, fuzzy rules and membership functions of the controller can be tuned automatically by the learning algorithm. Learning is based on the error evaluated by comparing the output of the ANFPSS and a desired PSS.
In a typical situation, a desired controller may either not be available, or the extensive input-output data required for training may not be easy to procure. A self-learning FLC [29] does not require another desired controller to obtain the training data. It is trained from the controlled plant output which in the case of the self-learning ANFPSS has been taken as the generator speed deviation.

In this approach, first a function approximator (or model) is required to represent the input-output behaviour of the plant. An adaptive network based fuzzy logic model, which has the same structure as the controller, is employed to model the plant. The function of this model is to compute the derivative of the model output with respect to its input by means of the back propagation process. Consequently, by propagating errors between the actual and the desired plant outputs, back through the model, error in the control signal can be calculated. The error in the control signal can be used to train the controller. A block diagram of the self-learning FLC, showing an adaptive network containing two subnetworks, the fuzzy controller and the plant model, is shown in Figure 7.

The training process for the controller starts from an initial state at t = 0. Then the FLC and the plant model generate the next states of control and $\Delta\omega$ at time t = h. The process continues till the plant state trajectory is determined based on the minimization of a performance index.

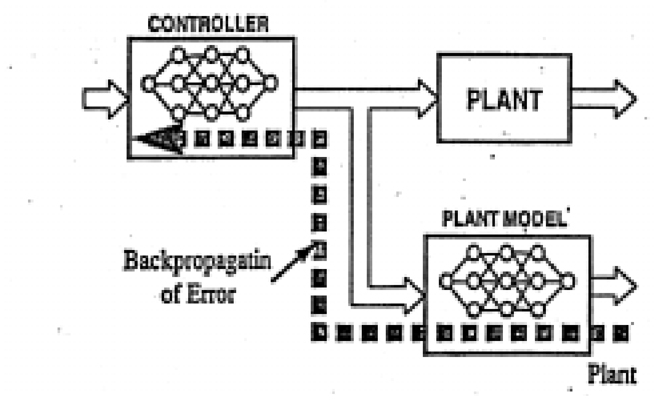


Figure 7. Error backpropagation through plant model

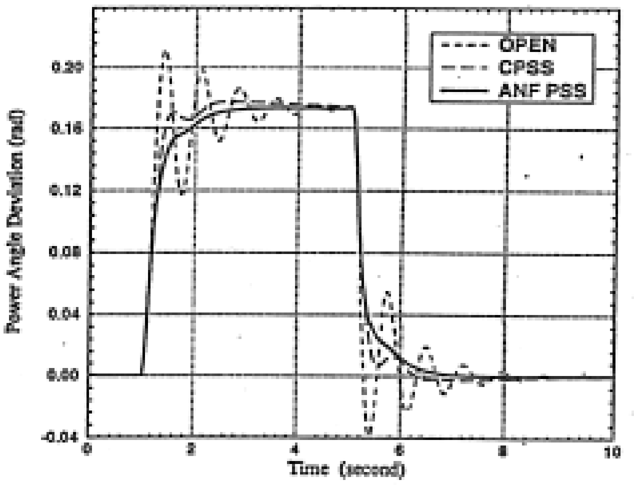


Figure 8. Response to a 0.20 pu step in torque under leading power factor condition

A number of studies have been performed to investigate the performance of the self-learning FLC structure employed as a self-learning ANF PSS on a single-machine infinite-bus system [30,34,35]. One illustrative result for a 0.2 pu step increase in torque under leading power factor conditions is shown in Figure 8.

F. Field Studies of a Fuzzy Logic Power System Stabilizer

Over the last several years, there has been work on the development of a fuzzy logic power system stabilizer to enhance the damping of generator oscillations as a joint research work between Kumamoto University and the Kyushu Electric Power Co. Through simulation studies and also experimental studies on a 5kVA laboratory system, several hydro units in the Kyushu Electric Power System, and the Analog Power System Simulator at the Research Laboratory of the Kyushu Electric Power Co., the excellent control performance of the proposed fuzzy logic power system stabilizer has been demonstrated. The fuzzy logic power system stabilizer manufactured by Toshiba Co. is now in service on a 90MVA hydro unit in the Kyushu Electric Power System. Further studies are now ongoing for the development of the integrated fuzzy logic generator controller and the fuzzy logic excitation controller.

This section summarizes the development of the fuzzy logic power system stabilizer (FLPSS) during the last several years. First, simulation studies have been performed to investigate the control performance and the robustness of the proposed FLPSS by using a simple one machine and infinite bus system and several multi-machine power systems as the study systems. Then, after setting up a personal computer (PC) based FLPSS, experimental studies were performed on a 5kVA one machine infinite bus laboratory system, and the Analog Power System Simulator at the Research Laboratory of the Kyushu Electric Power Co.. mainly to investigate the feasibility of the proposed FLPSS and also to demonstrate the excellent control performance compared with conventional PSSs. First site tests were performed successfully at one of the hydro power stations in the Kyushu Electric Power System in October 1992. Before the installation of the FLPSS as a real equipment, two years long term evaluation of the PC based prototypes of the FLPSS was performed from March 1994 to March 1996 on two hydro units with the rating of 20 to 30 MVA in the Kyushu Electric Power System.

Following the long term evaluation, the PC based FLPSS was installed on a 90MVA hydro unit in May 1997. After standard site tests and also disturbance tests, the PC based FLPSS has been in service since June 1997 through May 1999. The PC based FLPSS has been replaced to the FLPSS manufactured by Toshiba Corp. in May 1999. The manufacturer made FLPSS has been in service after the standard PSS tests. In addition, a fuzzy logic excitation control system is also briefly introduced as further studies. The damping of oscillations is further improved by applying the fuzzy logic excitation control system compared with the combination of the conventional automatic voltage regulator (AVR) and the FLPSS. The fuzzy logic excitation control system has been tested on the 5kVA laboratory system and also on the Analog Power System Simulator to demonstrate its better control performance.

F.1 Simulation Studies [31, 8,9]

Simulations were performed for a simple one machine infinite bus systems and also for several multi-machine power systems to demonstrate the robustness of the FLPSS. Through the simulations, the roles of the adjustable parameters were also clarified. In addition, it has been shown that the FLPSS provides better damping compared with widely used conventional PSSs(CPSS).

F.2 Experimental Studies on 5kVA Laboratory System [13]

To demonstrate the effectiveness of the proposed FLPSS, experiments were performed by using a laboratory system rated at 5 kVA, 220 VAC, and 60 Hz shown. Disturbances were added to the laboratory system by changing the length of the transmission line which connects the generator to a commercial power source. On the monitoring system, real time monitoring is available to check the FLPSS performance. Wider stable region is achieved by applying the three-dimensional FLPSS.

F.3 Site Tests in October 1992 [32]

First site tests were performed at the Itsukigawa Hydro Power Stations in the Kyusyu Electric Power System in October, 1992 before the replacement of a 5 MVA unit (Unit No. 1. Step changes of the AVR reference voltage, reactance switching, and faulty synchronization of the study unit were considered as the disturbances at the site. The FLPSS demonstrated the better performance comparing with the CPSS.

F.4 Evaluation on Analog Power System Simulator [10]

The control performance of the FLPSS was investigated on the Analog Power System Simulator at the Research Laboratory of the Kyushu Electric Power Company. for several multi-machine systems. Through the investigation, it was demonstrated that the FLPSS could damp multi-mode oscillations: low frequency global mode of oscillations and high frequency local mode of oscillations

F.5 Long Term Evaluation of PC Based Prototype [10]

The PC based prototype of the FLPSS is shown in Figure 9 including the monitoring unit (PQVF), the protection unit, and the uninterruptible power system (UPS). The PQVF monitors the real and the reactive powers, the terminal voltage, and the system frequency.

Figure 9. PC Based Prototype of FLPSS

The first prototype was installed on a hydro-unit (30.2MVA, 11kV, 600rpm) at the Kurokawa No. 1 Hydro Power Station on March 14, 1994. This unit has a brush-less AC exciter(160kW, 260V). According to the experimental results on the Analog Simulator, the adjustable parameters were set at the site. The maximum size of the stabilizing signal *Umax* was set to 0.05 pu because of the regulation for the unit.

The second prototype was installed on the unit (23.4MVA, 6.6kV, 200rpm) at the Kawabaru Hydro Power Station on May 26, 1994 after the site tests. This unit has also a brush-less AC exciter (155kW, 160V). The excitation system has a digital AVR, therefore, the tuning of the FLPSS parameters were performed at the site using step changes of the reference voltage at the operating point of 5 MW output. The maximum size of the stabilizing signal was also set to 0.05 pu.

Through the long term evaluation, the better performance of the FLPSS was demonstrated and the reliablity of the PC based FLPSS was also recognized.

F.6 Permanent Installation after Disturbance Tests

The same PC based FLPSS was installed on the 90 MW Unit 2 at the Hitotsuse Hydro Power Station in the Kyushu Electric Power System in May 1997. Unit 2 has a thyristor exciter. The fuzzy control parameters were tuned at the site for a 3 % step change of the AVR reference voltage. Disturbance tests were also performed in June 1997 before the actual utilization of the FLPSS. Figure 10 shows the configuration of the South-Kyushu Subsystem. The disturbance was added to the system by opening the 220 kV line at the location of A. Figure 11 and Figure 12 show the results of the disturbance tests. These figures illustrate the response of Unit 2 on which the FLPSS was installed.

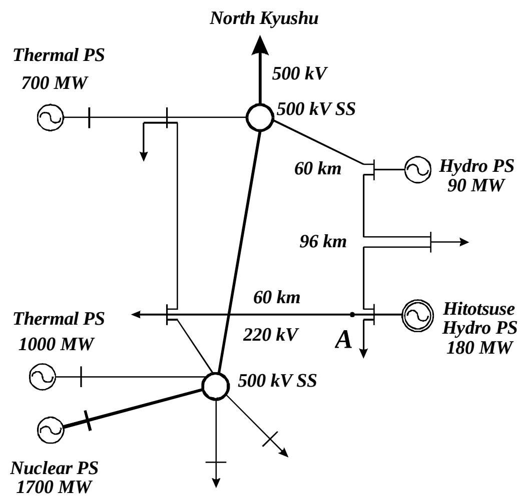


Figure 10. Configuration of South Kyushu Subsystem

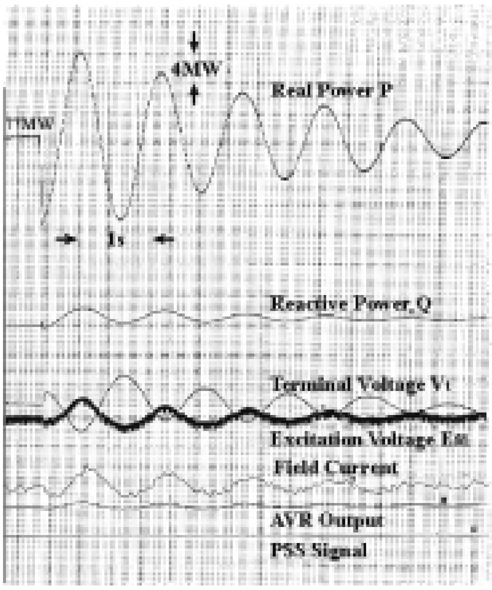


Figure 11. Response of Unit 2 without PSS

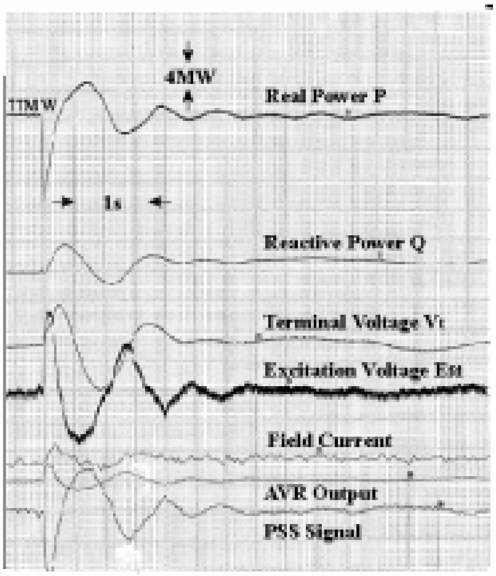


Figure 12. Response of Unit 2 with FLPSS

In Figure 11, the FLPSS on Unit 2 was locked. In Figure 12, the FLPSS was in service on Unit 2. As shown in there figures, the damping of Unit 2 is highly increased by the FLPSS. The maximum size of the stabilizing signal is set to 0.05 pu according to the regulation

The FLPSS has been in service since June 19, 1997. Since then, all the monitored data shows that the FLPSS provides increased damping to Unit 2. In addition, the PC based prototype of the FLPSS was replaced with the FLPSS manufactured by Toshiba Corp. in May 1999. After standard tests for the FLPSS, the FLPSS has been in service since then. The monitoring of the FLPSS performance has been continued for the future installation of the FLPSS on the other large-scale units.

G. Conclusions

In the previous sections a fuzzy system has been defined and it is shown how a fuzzy system can be used to approximate a controller. The determination of the membership functions and the fuzzy rule base is illustrated in two ways:

- the empirical way using linguistic sets and rules and human knowledge
- the self-organizing way using data samples and analysis.

Both approaches do not necessarily need a detailed state-space model of the plant. The advantage of the first approach is the use of heuristics and human knowledge. However the demonstration of stability for this type of controller is very tedious if not impossible.

Self-organizing controllers on the other hand fall into the class of adaptive controllers and the related stability issues can be explored with adaptive control techniques. Cited results show that fuzzy systems generalize the concept of function approximation. There is a class of fuzzy systems whose mapping is given as a Basis Function expansion of controller input and output data.

A lot of progress has been made concerning the application of fuzzy systems to power system control problems. For feasibility studies most authors experiment with empirical rules and data. However, a few projects, using self-organizing techniques, have been installed on a microprocessor and tested in a research lab environment either in academia or a utility. Recently, fuzzy controllers have achieved commercialization.

H. References

[1] Kundur, P. , *Power System Stability and Control*, McGraw-Hill, 1994.

[2] Wang, L-W, and Mendel, J.M. “Fuzzy Basis Functions, Universal Approximation and Orhtogonal Least-Squares Learning”, IEEE Transactions on Neural Networks, Vo. 3, No. 5, September 92, 807-814.

[3] Mendel, J.M., “Fuzzy Logic Systems for Engineering: A Tutorial”, Proceedings of the IEEE, Vol. 80, No. 3, March 1995, 345-377.

[4] Hassan, M., Malik, O., Hope, G.S., “Fuzzy Logic Based Stabilizer for a Synchronous Machine”, IEEE Transactions on Energy Conversion, Vol. 6, No. 3, June 1991, 407-413.

[5] Hassan, M., Malik, O., “Implementation and Laboratory Test Results for a Fuzzy Logic Based Self Tuned Power System stabilizer”, IEEE Transactions on Energy Conversion, Vol. 8, No. 2, June 1993, 221-228.

[6] El-Metwally, K., Hancock, G., Malik, O., “Implementation of a Fuzzy Logic PSS Using a Microcontroller and Experimental

Test Results", IEEE Transactions on Energy Conversion, Vol. 1, No. 1, March 1996, 91-96.
[7] Hiyama, T., Sameshima, T., and Lim, C.M., "A fuzzy Logic Stabilizers with Digital Compensator for Stability Enhancement of Multimachine Power Systems", Proc. of the Third Symposium on Expert System Applications to Power systems, Tokyo-Kobe, April 1991, 455-461.
[8] Hiyama, T., "Robustness of Fuzzy Logic Power System Stabilizers Applied to Multimachine Power System", IEEE Transactions on Energy Conversion, Vol. 9, No. 3, September 1994, 451-459.
[9] Hiyama, T., Kugimiya, M., Satoh, H., "Advanced PID Type Fuzzy Logic Power System Stabilizer", IEEE Transactions on Energy Conversion, Vol. 9, No. 3, September 1994, 514-520.
[10] Hiyama, T., Oniki, S., Nagashima, H., "Evaluation of Advanced Fuzzy Logic PSS on Analog Network Simulator and Actual Instalation on Hydro Generators", IEEE Transactions on Energy Conversion, Vol. 11, No. 1, March 1996, 125-131.
[11] Hiyama, T., Mishiro, M., Kihara, H., Ortmeyer, T., "Coordinated Fuzzy Logic Control for Series Capacitor Modules and PSS to Enhance Stability of Power System", IEEE Transactions on Power Delivery, Vol. 10, No. 2, April 1996, 1098-1104.
[12] Hiyama, T., Miyazaki, K., Satoh, H., "A Fuzzy Logic Excitation System for Stability Enhancement of Power Systems with Multimode Oscillations", IEEE Transactions on Energy Conversion, Vol. 11, No. 2, June 1996, 449-454.
[13] Hiyama, T., "Real Time Control of Micro Machine System Using Microcomputer Based Fuzzy Logic Power System Stabilizers", IEEE Transactions on Energy Conversion, Vol. 9, No. 4, December 1996, 724-731.
[14] Ortmeyer, T., Hiyama, T., "Frequency Response Characteristics of the Fuzzy Polar Power System Stabilizer", IEEE Transactions on Energy Conversion, Vol. 10, No. 2, June 1995, 333-338.
[15] Chang, H.C., Wang, M.H., "Neural Network Based Self Organizing Fuzzy Controller for Transient Stability of Multimachine Power Systems", IEEE Transactions on Energy Conversion, Vol. 10, No. 2, June 1995, 339-347.
[16] Handschin, E., Hoffmann, W., Reyer, F., Stephanblome, T., Schlucking, U., et. al., "A New Method Excitation Control Based on Fuzzy Set Theory", IEEE Transactions on Power systems, Vol. 9, No. 1, February 1996, 533-539.
[17] Hoang, P., Tomsovic, K., "Design and Analysis of An Adaptive Fuzzy Power System Stabilizer", IEEE Transactions on Energy Conversion, Vol. 11, No. 2, June 1996, 455-461.
[18] Noroozian, M., Andersson, G., Tomsovic, K., "Robust, Near Time Optimal Control of Power System Oscillations with Fuzzy Logic", IEEE Transactions on Power Delivery, Vol. 11, No. 1, January 1996, 393-400.
[19] Park, Y., Moon, U., Lee, K.W., "A Self-Organizing Power System Stabilizer Using Fuzzy Autoregressive Moving Average (Farma) Model", IEEE Transactions on Energy Conversion, Vol. 11, No. 2, June 1996, 442-448.
[20] Toliyat, H., Sadeh, J., Ghazi, R., "Design of Augmented Fuzzy Logic Power System Stabilizers to Enhance Power Systems Stability", IEEE Transactions on Energy Conversion, Vol. 1,, No. 1, March 1996, 97-103.
[21] Hiyama, T., Mishiro, M., Kihara, H., Ortmeyer, T., "Fuzzy Logic Switching of Thyristor Controlled Braking Resistor Considering Coordination with SVC", IEEE Transactions on Power Delivery, Vol. 10, No. 4, October 1996, 2020-2026.
[22] Hoang, T.C. and El-Sharkawi, M., "High Performance Speed and Position Tracking of Induction Motors using Multi-layer Fuzzy Control", IEEE Transactions on Energy Conversion, Vol. 11, No. 2, June 1996, 353-358.
[23] Tang, Y., Xu, L., "Fuzzy Logic Application for Intelligent Control of a Variable Speed Drive", IEEE Transactions on Energy Conversion, Vol. 9, No. 4, December 1994, 679-685.
[24] Tang, Y., Xu, L., "Vector Control and Fuzzy Logic Control of Doubly Fed Variable Speed Drives with DSP Implementation", IEEE Transactions on Energy Conversion, Vo. 10, No. 4, December 1996, 661-668.
[25] Hilloowala, R, Sharaf, A., "A Rule Based Fuzzy Logic Controller for a PWM Inverter in a Stand Alone Wind Energy Conversion Scheme", IEEE Transactions on Industrial Applications, Vol. 32, No. 1, Jan/Feb. 1996, 57-65.
[26] Lee, C.C., "Fuzzy Logic in Control Systems: Fuzzy Logic Controller - Parts I and II", IEEE Transactions on Systems, Man and Cybernetics, Vol. 20, 1990, 404-435.
[27] Jang, J.G., "ANFIS: Adaptive-Network-Based Fuzzy Inference System", IEEE Transactions on Systems, Man and Cybernetics, Vol. 23, 1993, 665-684.
[28] Hariri, A., and O.P. Malik, "Self-learning adaptive-network-based fuzzy logic power system stabilizer", Proc. Intl. Conf. on `Intelligent Systems Applications to Power Systems' ", Orlanda, Florida, Jan. 28-Feb. 2, 1996, pp. 299-303.
[29] Jang, J.G., "Self-learning Fuzzy Controllers Based on Temporal Backpropagation", IEEE Transactions on Neural Networks, Vol. 3, 1992, 714-723.
[30] Hariri, A. and Malik, O.P., "A Fuzzy Logic Based Power System Stabilizer with Learning Ability". IEEE Transactions on Energy Conversion, Vol. 11, No. 6, Dec. 1996, 721-727.
[31] Hiyama, T., and Sameshima, T., "Fuzzy logic control scheme for on-line stabilization of multi-machine power system", Fuzzy Sets and Systems, Vol. 39, 1991, pp.181-194.
[32] Hiyama, T., Oniki, S., and Nagashima, H., "Experimental studies on micro-computer based fuzzy logic stabilizer", Proceedings of the 2nd International Forum on Application of Neural Network to Power Systems, pp.212-217, 1993.
[33] El-Metwally, KA. and Malik, O.P., "Applications of fuzzy stabilizers in a multi-machine power system environment," IEE Proceedings, Generation, Transmission and Distribution, Vol. 143, No. 3, May 1996, pp.263-268.
[34] Hairi, A. and Malik, O.P., "A self-learning fuzzy stabilizer for a synchronous machine," International Journal of Engineering Intelligent Systems, Vol. 5, No. 3, 1997, pp. 157-162.
[34] Hairi, A. and Malik, O.P., "Fuzzy logic power system stabilizer based on genetically optimized adaptive network," Fuzzy Sets and Systems, Vol. 102, No. 1, 1999, pp.31-40.

Chapter 4 Control Design and Stability

A. Control Design Techniques

When fuzzy systems are used as controllers, they are called *fuzzy controllers*. If fuzzy systems are used to model the process and controllers are designed based on the model, then resulting controllers also are called fuzzy controllers. Therefore, fuzzy controllers are nonlinear controllers with a special structure. Fuzzy control has represented the most successful applications of fuzzy theory to practical problems.

Fuzzy control can be classified into static fuzzy control and adaptive fuzzy control. In static fuzzy control, the structure and parameters of the fuzzy controller are fixed and do not change during real-time operation. On the other hand in adaptive fuzzy control, the structure and/or parameters of the fuzzy controller change during real-time operation. Fixed fuzzy control is simpler than adaptive fuzzy control, but requires more knowledge of the process model or heuristic rules. Adaptive fuzzy control, on the other hand, is more expensive to implement, but requires less information and may perform better.

A.1 Fixed Fuzzy Controller Design

Fuzzy control and conventional control have similarities and differences. They are similar in the sense that they must address the same issues that are common to any control problem, such as stability and performance. However, there is a fundamental difference between fuzzy control and conventional control. Conventional control starts with a mathematical model of the process and controllers are designed based on the model. Fuzzy control, on the other hand, starts with heuristics and human expertise (in terms of fuzzy *IF-THEN* rules) and controllers are designed by synthesizing these rules. That is, the information used to construct the two types of controllers is different; see Fig. 1. Advanced fuzzy controllers, however, can make use of both heuristics and mathematical models.

For many practical problems, it is difficult to obtain an accurate yet simple mathematical model, but there are human experts who can provide heuristics and rule-of-thumb that are very useful for controlling the process. Fuzzy control is most useful for these kinds of problems. If the mathematical model of the process is unknown, we can design fuzzy controllers in a systematic manner that guarantees certain key performance criteria.

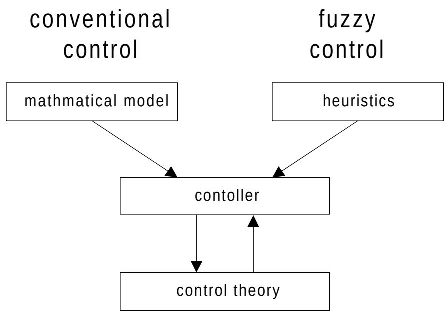


Fig. 1 Fuzzy control and conventional control.

The design techniques for fuzzy controllers can be classified into the trial-and-error approach and the theoretical approach [1]. In the trial-and-error approach, a set of fuzzy IF-THEN rules are collected from human experts or documented knowledge base, and the fuzzy controllers are constructed from these fuzzy IF-THEN rules. The fuzzy controllers are tested in the real system and if the performance is not satisfactory, the rules are fine-tuned or redesigned in a number of trial-and-error cycles until the performance is satisfactory. In theoretical approach, the structure and parameters of the fuzzy controller are designed in such a way that certain performance criteria are guaranteed. Both approaches, of course, can be combined to give the best fuzzy controllers.

<u>Trial-And-Error Approach:</u>

The trial-and-error approach to fuzzy controller design can be summarized in the following steps:

1. Select state and control variables. The state variables should characterize the key features of the system and the control variables should be able to influence the states of the system. The state variables are the inputs to the fuzzy controller and the control variables are the output of the fuzzy controller.
2. Construct IF-THEN rules between the state and control variables. The formulation of these rules can be achieved in two different heuristic approaches. The most common approach is the linguistic verbalization of human experts. Another approach is to interrogate experienced experts or operators using a carefully organized questionnaire.
3. Test the fuzzy IF-THEN rules in the system. The closed-loop system with the fuzzy controller is run and if the performance is not satisfactory, fine tune or

redesign the fuzzy controller and repeat the procedure until the performance is satisfactory.

The resulting fuzzy IF-THEN rule can be in the following two types:

Type I: IF x_1 is A_1^i AND ... AND x_n is A_n^i,

THEN u is B^j.

Type II:IF x_1 is A_1^i AND ... AND x_n is A_n^i,

THEN u is $c_0^i + c_1^i x_1 + \ldots + c_n^i x_n$.

In Type I, both the antecedent and consequent have linguistic variables, $A_k^i, k = 1,2,\ldots,n$ and B^j, respectively. On the other hand in Type II, the consequent is a parameterized function of the input to the fuzzy controller, or the state variables [2]. Comparing the two types, the THEN part of the rule is changed from a linguistic description to a simple mathematical formula. This change makes it easier to combine the rules. In fact, Type II, the Takagi-Sugeno system, is a weighted average of the rules in the THEN parts of the rules. This framework is useful in tuning the rules mathematically [3]. Type II, on the other hand has drawbacks: (i) its THEN part is a mathematical formula and therefore may not provide a natural framework to represent human knowledge, and (ii) there is not much freedom left to apply different principles in fuzzy logic, so that the versatility of fuzzy systems is not fully represented in this framework.

Theoretical Approach:

Knowing the mathematical model of a system is not a necessary condition for designing fuzzy controllers. However, in order to analyze the performance of the closed-loop fuzzy control system theoretically, we need to have some knowledge on the model of the system. This approach assumes a mathematical model for the system, so that mathematical analysis can be performed to establish the properties of the designed system.

Theoretical approach can be classified into the following categories:

1. Stable controller design
2. Optimal controller design
3. Sliding mode controller design
4. Supervisory controller design
5. Fuzzy system model-based controller design

1) Stable Controller Design - For control systems, stability is the most important requirement. Conceptually, there are two classes of stability: Lyapunov stability and input-output stability. We assume that the system is represented as a linear system and the fuzzy controller is connected in the feedback path as shown in Fig. 2.

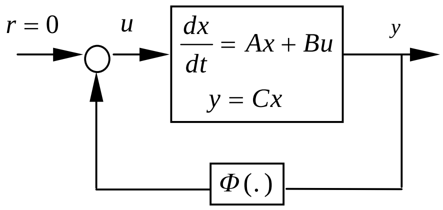


Fig. 2. Closed-loop fuzzy control system.

The overall system is described by the following equations:

$$\frac{dx(t)}{dt} = Ax(t) + Bu(t), \tag{1}$$

$$y(t) = Cx(t), \tag{2}$$

$$u(t) = -\Phi[y(t)], \tag{3}$$

where $x(t)$, $u(t)$, $y(t) \in \Re$ and Φ is a fuzzy system. Then we have the following exponential stability theorem:

Theorem 1.1[4]: Consider the system (1)-(2), and suppose that (a) all eigenvalues of A lie in the open left half of the complex plane, (b) the system is controllable and observable, and (c) the transfer function of the system is strictly positive real. If the nonlinear function Φ satisfies $\Phi(0) = 0$ and

$$y\Phi(y) \geq 0, \forall y \in R \tag{4}$$

then the equilibrium point $x = 0$ of the closed-loop system (1)-(3) is globally exponentially stable.

Conditions (a)-(c) in the theorem are imposed on the system under control, not on the controller. They are simply requiring that the open-loop system is stable and well-behaved. Conceptually, these systems are not difficult to control, and the conditions on the fuzzy controller are not very strong. The theorem guarantees that if we design a fuzzy controller $\Phi(y)$ that satisfies $\Phi(0) = 0$ and (4), then the closed-loop system is globally exponentially stable, provided that the system under control is linear and satisfies conditions (a)-(c). This leads to the design of a stable fuzzy logic controller:

Step 1. Define 2N+1 fuzzy sets A^l on the output space [-1, 1]that are normal, consistent, and complete with the triangular membership functions as shown in Fig. 3, where the first N fuzzy sets cover the negative interval [-1, 0], the last N fuzzy sets cover the positive interval [0,1], and the center of the middle fuzzy set for l=N+1 is at zero.

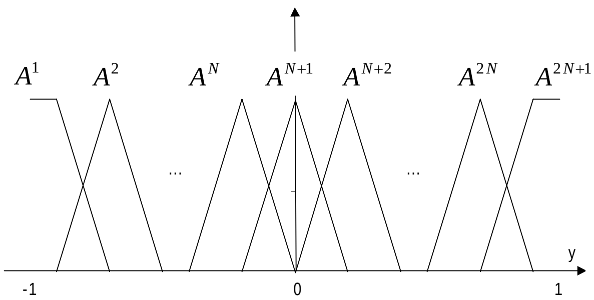


Fig. 3. Fuzzy controller membership functions.

Step 2. Define the following 2N+1 fuzzy IF-THEN rules:

$$\text{IF } y \text{ is } A^l \text{, THEN } u \text{ is } B^l \tag{5}$$

where l = 1, 2, ... , 2N+1, and the centers of fuzzy sets B^l are chosen such that

$$\bar{u}^i \begin{cases} \leq 0 & for \quad l = 1, \ldots, N \\ = 0 & for \quad l = N+1 \\ \geq 0 & for \quad l = N+2, \ldots, 2N+1 \end{cases} \tag{6}$$

Step 3. Design the fuzzy controller from the IF-THEN rules (6) using the product inference engine, singleton fuzzifier, and center average defuzzyfier; i.e., the designed fuzzy controller is

$$u = -\Phi(y) = -\frac{\sum_{l=1}^{2N+1} \bar{u}^l \mu_{A^l}(y)}{\sum_{l=1}^{2N+1} \mu_{A^l}(y)} \tag{7}$$

where $\mu_{A^l}(y)$ are shown in Fig. 3 and $\bar{u}^l$ satisfy (6).

The above design steps imply that in designing the fuzzy controller, we do not need to know the system model. Also, there is much freedom in choosing the parameters of the fuzzy controller.

When a nonzero input is applied to the fuzzy controlled system shown in Fig. 2, the input-output stability can be established following the following theorem:

Theorem 1.1.2 [4]: Consider the system in Fig. 2 and suppose that the nonlinear controller $\Phi(y)$ is globally Lipschitz continuous, that is,

$$\left|\Phi(y_1) - \Phi(y_2)\right| \leq K\left|y_1 - y_2\right|, \forall y_1, y_2 \in R \tag{8}$$

for some constant *K*. If the eigenvalues of *A* lie in the open left-half complex plane, then the forced closed-loop system in Fig. 3 is L_p-stable for all $p \in [1, \infty]$.

It can be shown that the fuzzy controller $\Phi(y)$ of (7) is continuous, bounded, and piecewise linear, and hence satisfies the Lipschitz condition (8) [1]. Thus, the closed-loop fuzzy control system in Fig. 2 is L_p- stable for all $p \in [1, \infty]$.

The stable fuzzy controller design can be easily extended to multi-input multi-output systems with *m* input/output variables. The IF-THEN rule (5) is generalized for the *j*'th group (*j* = 1,2,...,*m*) to the set of $\prod_{i=1}^m (2N_i + 1)$ rules:

$$\text{IF } y_1 \text{ is } A_1^{l_1} \text{ and ... and } y_m \text{ is } A_m^{l_m} \text{ THEN } u_j \text{ is } B_j^{l_1 \ldots l_m} \tag{9}$$

where l_i = 1, 2, ... , $2N_i$+1, *i* = 1,2,...,*m* and the centers of fuzzy sets $B_j^{l_1 \ldots l_m}$ are chosen such that

$$\bar{u}_j^{l_1 \ldots l_m} \begin{cases} \leq 0 & for \quad l_j = 1, \ldots, N_j \\ = 0 & for \quad l_j = N_j + 1 \\ \geq 0 & for \quad l_j = N_j + 2, \ldots, 2N_j + 1 \end{cases} \tag{10}$$

where l_i for *i* = 1,2,...,*m* can take any values from {1,2,...,$2N_i$+1}. The resulting fuzzy controller is

$$u_j = -\Phi_j(y) = -\frac{\sum_{l_1=1}^{2N+1} \cdots \sum_{l_m=1}^{2N+1} \bar{u}^{l_1 \ldots l_m} (\prod_{i=1}^m \mu_{A_i^{l_i}}(y_i))}{\sum_{l_1=1}^{2N+1} \cdots \sum_{l_m=1}^{2N+1} (\prod_{i=1}^m \mu_{A_i^{l_i}}(y_i))} \tag{11}$$

where *j* = 1,2,...,*m*.

2) Optimal Controller Design - The stable controller determines the range for fuzzy controller parameters for which the stability is guaranteed; however, it does not show how to determine specific values of the parameters. Optimal controller, on the other hand, determines the specific values of the fuzzy controller parameters such that certain performance criterion is minimized.

From the stable fuzzy controller (11), we define the fuzzy basis functions $b(x) = (b_1(x), \ldots, b_N(x))^T$ as

$$b_l(x) = -\frac{\prod_{i=1}^n \mu_{A_i^{l_i}}(x_i)}{\sum_{l_1=1}^{2N+1} \cdots \sum_{l_m=1}^{2N+1} (\prod_{i=1}^n \mu_{A_i^{l_i}}(x_i))} \tag{12}$$

where l_i = 1,2,...,$2N_i$+1, *l* = 1,2,...,*N* and $N = \prod_{i=1}^n (2N_i + 1)$

Define an $m \times N$ parameter matrix Θ as

$$\Theta = \begin{bmatrix} -\Theta_1^T \\ \cdots \\ -\Theta_m^T \end{bmatrix} \tag{13}$$

where $\Theta_j^T \in R^{1 \times N}$ consists of the N parameters $\bar{u}_j^{l_1 \ldots l_n}$ for l_i = 1,2, ...,$2N_i$+1 in the same ordering as $b_l(x)$ for l = 1,2,...,*N*. Then the fuzzy controller $u = (u_1, \ldots, u_m)^T$ can be expressed as

$$u = \Theta b(x)\,. \tag{14}$$

Substituting (14) into (1), the closed-loop system is obtained as

$$\dot{x}(t) = Ax(t) + B\Theta(t)b(x(t))\,, \tag{15}$$

where the parameter matrix is assumed to be time-varying. The optimal control problem can then be formulated to minimize the following performance criterion

$$J = \tfrac{1}{2}x^T(T)Sx(T) + \tfrac{1}{2}\int_0^T [x^T Qx + b^T(x)\Theta^T R\Theta b(x)]dt \tag{16}$$

Thus, the problem of designing the optimal fuzzy controller becomes the problem of determining the optimal $\Theta(t)$, which can be solved by applying the Pontryagin maximum principle [5]. Specifically, by minimizing the Hamiltonian function

$$H(x, p, \Theta) = x^T Qx + b^T\Theta^T R\Theta b + p^T(Ax + B\Theta b) \tag{17}$$

the optimal fuzzy controller parameter matrix is obtained as

$$\Theta^*(t) = -\tfrac{1}{2}R^{-1}B^T p^* b^T(x^*)[b(x^*)b^T(x^*)]^{-1}\,, \tag{18}$$

where x and p are the solution of the Hamiltonian system:

$$\begin{aligned} \dot{x}^* &= \frac{\partial H(x^*, p^*, \Theta^*)}{\partial p}; \qquad x(0) = x_0 \\ \dot{p}^* &= -\frac{\partial H(x^*, p^*, \Theta^*)}{\partial x}; \qquad p(T) = Sx(T) \end{aligned} \tag{19}$$

and thus the optimal fuzzy controller is

$$u^* = \Theta^*(t)b(x)\,. \tag{20}$$

We note that the optimal fuzzy controller (20) is a state feedback controller with time-varying parameters.

3) Sliding Mode Controller Design - Sliding mode control is a robust control method for nonlinear and uncertain dynamic systems [6,7]. It can be applied in the presence of model uncertainties and parameter disturbances, provided that the bounds of these uncertainties and disturbances are known. In many respects, the sliding mode control is similar to fuzzy control [8,9].

Consider a SISO nonlinear system

$$x^{(n)} = f(\mathbf{x}) + u \tag{21}$$

where $u \in R$ is the control input, $x \in R$ is the output, and $\mathbf{x} = (x, \dot{x}, \cdots, x^{(n-1)})^T \in R^n$ is the state vector. The uncertainty of the model is bounded by a known function:

$$f(\mathbf{x}) = \hat{f}(\mathbf{x}) + \Delta f(\mathbf{x}) \tag{22}$$

and

$$|\Delta f(\mathbf{x})| \le F(\mathbf{x}) \tag{23}$$

where $\Delta f(\mathbf{x})$ is unknown but $\hat{f}(\mathbf{x})$ and $F(\mathbf{x})$ are known. The control objective is to determine a feedback control $u = u(\mathbf{x})$ such that the state $\mathbf{x}$ will follow the desired state $\mathbf{x}_d = (x_d, \dot{x}_d, \cdots, x_d^{(n-1)})^T$, i.e., the tracking error

$$\mathbf{e} = \mathbf{x} - \mathbf{x}_d = (e, \dot{e}, \cdots, e^{(n-1)})^T \tag{24}$$

should converge to zero, where $e = x - x_d$.

Define a scalar function

$$\begin{aligned} s(\mathbf{x}, t) &= (\frac{d}{dt} + \lambda)^{n-1} e \\ &= e^{(n-1)} + C_{n-1}^1 \lambda e^{(n-2)} + C_{n-1}^2 \lambda^2 e^{(n-3)} + \cdots + \lambda^{n-1} e \end{aligned} \tag{25}$$

where λ is a positive constant. Then

$$s(\mathbf{x}, t) = 0 \tag{26}$$

defines a time-varying *sliding surface* $S(t)$ in the state space R^n. The equation (26) has a unique solution $e(t) = 0$ for the zero initial condition $e(0) = 0$. Thus, the tracking control problem is equivalent to keeping the scalar function $s(\mathbf{x}, t)$ at zero. This can be achieved by the *sliding condition*

$$\tfrac{1}{2}\tfrac{d}{dt}s^2 \le -\eta|s| \tag{27}$$

when the state is outside of $S(t)$, where η is a positive constant.

Consider, for example, a second order system (n=2), then the sliding surface $S(t)$ is

$$s(\mathbf{x}, t) = \dot{e} + \lambda e = \dot{x} + \lambda x - \dot{x}_d - \lambda x_d = 0 \tag{28}$$

which is a strait line in the $x - \dot{x}$ phase plane as shown in Fig. 4.

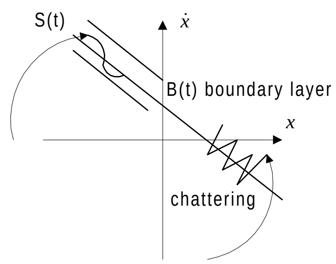


Fig. 4. Sliding surface, chattering, and boundary layer.

It can be shown that the sliding condition (27) is satisfied if we choose the control to the second order system as

$$u = -\hat{f}(\mathbf{x}) + \ddot{x}_d - \lambda\dot{e} - [\eta + F(\mathbf{x})]sgn(s)\,. \tag{29}$$

This sliding control law, however, is discontinuous across the sliding surface and, since the control switching cannot be perfect, it causes chattering in when implemented, as shown in Fig. 4. In order to eliminate chattering, a thin boundary layer around the sliding surface,

$$B(t) = \{\mathbf{x}: |s(\mathbf{x},t)| \le d\}, \qquad (30)$$

is introduced so that the control changes continuously within this boundary layer, Fig. 4, where d and $\varepsilon = d / \lambda^{n-1}$ are the thickness and width of the boundary layer, respectively. If the control law satisfies the sliding condition (27) outside of the boundary layer $B(t)$, then the error tracking is guaranteed to be within the precision of ε. Thus, a smooth controller can be designed that does not need to switch discontinuously across the sliding surface. For the second-order system, this is achieved by modifying the control law (29) as

$$u = -\hat{f}(\mathbf{x}) + \ddot{x}_d - \lambda\dot{e} - [\eta + F(\mathbf{x})]sat(s/d) \qquad (31)$$

where the saturation function is defined as

$$sat(s/d) = \begin{cases} -1 & if \quad s/d \le -1 \\ s/d & if \quad -1 \le s/b \le 1 \\ 1 & if \quad s/d \ge 1 \end{cases} \qquad (32)$$

The fuzzy controller can now be designed by viewing the smooth sliding controller (31) as the center of the output fuzzy set, $g(e,\dot{e})$.

4) Supervisory Controller Design - The fuzzy control systems discussed above are all single-loop (or single-level) controllers. For complex systems, the single-loop control systems may not effectively achieve the control objectives, and multi-level control structure becomes a necessity. The low level controllers perform fast direct control and the higher-level controllers perform low-peed supervision. There are two types of two-level controls: a) low-level fuzzy control and high-level nonfuzzy supervisory control, and b) low-level nonfuzzy control and high-level fuzzy supervisory control, Fig. 5.

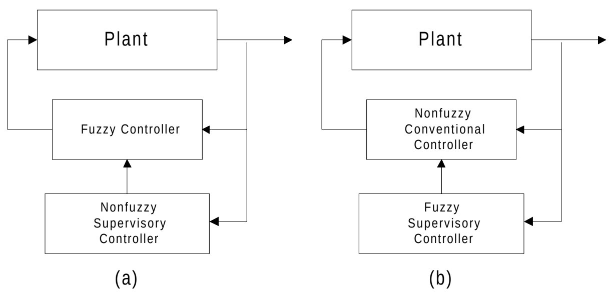


Fig. 5. Two-level fuzzy control systems: (a) fuzzy-local nonfuzzy-supervisory control, (b) nonfuzzy-local fuzzy-supervisory control.

In the two-level control system in Fig. 5(a), the fuzzy controller can be designed without considering stability and the supervisory controller can be designed to ensure the stability and other performance requirements [10]. In this way, there is much freedom in choosing the fuzzy controller parameters and consequently, the design of the fuzzy controller is simplified. Since the fuzzy controller is to perform the main control action, the supervisory control will play a supplementary action, that is, if the fuzzy controller works well, the supervisory control will be idle; if the fuzzy control system tends to be unstable, the supervisory controller starts working to enforce stability.

Consider the nonlinear system

$$x^{(n)} = f(x, \dot{x}, \ldots, x^{(n-1)}) + g(x, \dot{x}, \ldots, x^{(n-1)})u \qquad (33)$$

where $x \in R$ is the output, $u \in R$ is the control, $\mathbf{x} = (x, \dot{x}, \ldots, x^{(n-1)})^T$ is the state vector, and f and g are unknown nonlinear functions with $g > 0$ assumed. This type of system can be linearized with nonlinear feedback and a stable linear controller con be designed [11].

Suppose a fuzzy controller is already designed and we want to guarantee the stability of the closed-loop system in the sense that the state x is uniformly bounded, i.e., $|\mathbf{x}(t)| \le M_x, \forall t > 0$, where M_x is a constant. This can be achieved by supplementing a supervisory controller to the fuzzy controller:

$$u = u_{fuzz}(\mathbf{x}) + I^* u_s(\mathbf{x}) \qquad (34)$$

where the indicator function $I^* = 1$ if $|\mathbf{x}| \ge M_x$ and $I^* = 0$ if $|\mathbf{x}| \le M_x$. The goal is now to design the supervisory controller u_s such that $|\mathbf{x}| \le M_x$ for all t > 0. The closed-loop system then becomes

$$x^{(n)} = f(\mathbf{x}) + g(\mathbf{x})u_{fuzz}(\mathbf{x}) + g(\mathbf{x})I^* u_s(\mathbf{x}) \qquad (35)$$

The feedback linearization controller for the system (33) is given by

$$u_{FL} = \frac{1}{g(\mathbf{x})}[-f(\mathbf{x}) - \mathbf{k}^T\mathbf{x}] \qquad (36)$$

where $\mathbf{k} = (k_n, \ldots, k_1)^T \in R^n$ is such that all roots of the polynomial $s^n + k_1 s^{n-1} + \cdots + k_n$ are in the left-half complex plane. The system (35) is then rewritten as

$$x^{(n)} = -\mathbf{k}^T\mathbf{x} + g(\mathbf{x})[u_{fuzz} - u_{FL} + I^* u_s] \qquad (37)$$

or, in the matrix form,

$$\dot{\mathbf{x}} = \Lambda\mathbf{x} + \mathbf{b}[u_{fuzz} - u_{FL} + I^* u_s] \qquad (38)$$

where

$$\Lambda = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 \\ -k_n & -k_{n-1} & \cdots & \cdots & \cdots & \cdots & -k_1 \end{bmatrix}, \ \mathbf{b} = \begin{bmatrix} 0 \\ \cdots \\ 0 \\ g \end{bmatrix} \quad (39)$$

The supervisory controller u_s can be designed to guarantee $|\mathbf{x}| \leq M_x$ for all t > 0 by introducing a Lyapunov function

$$V = \frac{1}{2}\mathbf{x}^T P \mathbf{x} \quad (40)$$

where P is a symmetric positive definite matrix satisfying the Lyapunov equation

$$\Lambda^T P + P\Lambda = -Q \quad (41)$$

where Q is specified by the designer. Using (39) and (41), we have

$$\begin{aligned} \dot{V} &= -\frac{1}{2}\mathbf{x}^T Q\mathbf{x} + \mathbf{x}^T P\mathbf{b}[u_{fuzz} - u_{FL} + u_s] \\ &\leq \left|\mathbf{x}^T P\mathbf{b}\right|\left(\left|u_{fuzz}\right| + \left|u_{FL}\right|\right) + \mathbf{x}^T P\mathbf{b}u_s \end{aligned} \quad (42)$$

The supervisory controller u_s can be designed such that $\dot{V} \leq 0$ by choosing

$$u_s = -sign(\mathbf{x}^T P\mathbf{b})\left[\frac{1}{g_L}\left(f^U + \left|\mathbf{k}^T\mathbf{x}\right|\right) + \left|u_{fuzz}\right|\right] \quad (43)$$

where f^U and g_L are the upper and lower bounds of f and g, respectively.
Since the indicator function is a step function it may cause chattering at the boundary $|x| = M_x$, and this can be avoided by defining a continuous function

$$I^* = \begin{cases} 0 & |\mathbf{x}| \leq a \\ \dfrac{|\mathbf{x}| - a}{M_x - a} & a \leq |\mathbf{x}| \leq M_x \\ 1 & |\mathbf{x}| \geq M_x \end{cases} \quad (44)$$

Fuzzy controller can be used to tune the gains of conventional proportional-integral-derivative (PID) controllers. The transfer function of a PID controller has the following form:

$$G(s) = K_P + K_I / s + K_D s \quad (45)$$

where K_P, K_I, and K_D are the proportional, integral, and derivative gains, respectively. An equivalent form of the PID controller in time-domain is

$$u(t) = K_P[e(t) + \frac{1}{T_I}\int_0^t e(\tau)d\tau + T_D\dot{e}(t)] \quad (46)$$

where $e(t)$ is the error between the set point and the response of the system, and $T_I = K_P/K_I$ and $T_D = K_D/K_P$ are the integral and derivative time constants, respectively.

The PID gains are usually tuned by experienced experts based on heuristics. This is where fuzzy IF-THEN rules can be used. The PID gains can be tuned by analyzing the responses of the system on-line [12]. The input to the fuzzy system can be $e(t)$ and $\dot{e}(t)$, and outputs of the fuzzy system can be the PID gains. The PID gains are usually normalized, for example, as

$$\overline{K}_P = \frac{K_P - K_{P\min}}{K_{P\max} - K_{P\min}} \quad (47)$$

Then the fuzzy IF-THEN rules can be of the form:

$$IF\ e(t)\ is\ A^l\ and\ \dot{e}(t)\ is\ B^l, THEN\ \overline{K}_P\ is\ C^l, \overline{K}_D\ is\ D^l, \overline{K}_I\ is\ E^l \quad (48)$$

where A^l, B^l, C^l, D^l, and E^l are fuzzy sets, and l = 1, 2, ..., M.

5) Fuzzy System Model-Based Controller Design - The fuzzy control systems discussed above assumed that the systems under control are represented by ordinary linear or nonlinear dynamic system models. In many practical problems, however, human experts may provide linguistic descriptions about the system that can be combined into a model of the system; this model is called a *fuzzy system model*. There are two types of fuzzy system models, the Takagi-Sugeno-Kang (TSK) fuzzy system model [13-15] and the fuzzy-autoregressive-moving-average (FARMA) model [16,17].

The Takagi-Sugeno-Kang Fuzzy System Model: The Takagi-Sugeno-Kang (TSK) fuzzy system was proposed as an alternative to the usual fuzzy systems. The TSK fuzzy system is made of the following rules:

$$IF\ x_1\ is\ C_1^l\ and \cdots and\ x_n\ is\ C_n^l, THEN\ y^l = c_0^l + c_1^l x_1 + \cdots + c_n^l x_n \quad (49)$$

where C_i^l are fuzzy sets, c_i^l are constants, and l = 1, 2, ..., M. Thus, the antecedent parts of the rules are the same as in the usual fuzzy IF-THEN rules, but the consequent parts are linear combinations of the input variables. Given an input $x = (x_1, \ldots, x_n)^T \in U \subset R^n$, the output $f(x) \in V \subset R$ of the TSK fuzzy system is computed as the weighted average of the outputs, i.e.,

$$f(x) = \frac{\sum_{l=1}^M y^l w^l}{\sum_{l=1}^M w^l} \quad (50)$$

where the weights w^l are computed as

$$w^l = \prod_{i=1}^n \mu_{C_i^l}(x_i) \quad (51)$$

The fuzzy system is a mapping from $U \subset R^n$ and $V \subset R$, and the output is a piece-wise linear function of the input variables, where the change from one piece to another is smooth rather than abrupt. If $c_i^l = 0$ for i = 1, 2, ..., n and c_0^l equals the center $\bar{y}^l$ of the fuzzy set B^l in the usual fuzzy IF-THEN rules, then the TSK fuzzy system is identical to the fuzzy system with product inference, singleton fuzzifier, and center average defuzzifier.

If the output of a TSK fuzzy system appears as one of its inputs, a *dynamic* TSK fuzzy system is obtained:

$$\begin{aligned} &IF\ x(k)\ is\ A_1^p and \cdots and\ x(k-n+1)\ is\ A_n^p and\ u(k)\ is\ B^p \\ &THEN\ x^p(k+1) = a_1^p x(k) + \cdots + a_n^p x(k-n+1) + b^p \end{aligned} \tag{52}$$

where A_i^p and B^p are fuzzy sets, a_i^p and b^p are constants, p = 1, 2, ..., N, $u(k)$ is the input to the system, and $\mathbf{x}(k) = (x(k), x(k-1), \ldots, x(k-n+1))^T \in R^n$ is the state vector of the system. The output of the dynamic TSK fuzzy system is computed as

$$x(k+1) = \frac{\sum_{p=1}^{N} x^p(k+1) v^p}{\sum_{p=1}^{N} v^p} \tag{53}$$

where the weights v^p are computed as

$$v^p = \prod_{i=1}^{n} \mu_{A_i^p}[x(k-i+1)] \mu_{B^p}[u(k)] \tag{55}$$

This dynamic TSK fuzzy system is used to model the plant under control, and the TSK fuzzy control (50) is used to control the plant, Fig. 6.

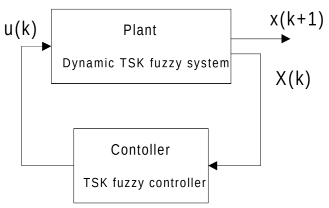


Fig. 6. Fuzzy control with fuzzy system model.

It remains to determine the stability of the closed-loop fuzzy control system, in other words, the controller parameters, c_i^l and $\mu_{C_i^l}$ in (49), need to be designed to guarantee the stability of the fuzzy controlled system. Assuming that the parameters of the dynamic TSK fuzzy system model (52) are known, the following gives the sufficient condition for stability:

Theorem 1.5.1[15]: The dynamic TSK fuzzy system is globally asymptotically stable if there exists a common positive definite matrix P such that

$$A_{lp}^T P A_{lp} - P < 0 \tag{56}$$

for all l = 1, 2, ..., M and p = 1, 2, ..., N, where

$$A_{lp} = \begin{bmatrix} a_1^p + b^p c_1^l & a_2^p + b^p c_2^l & \cdots & a_{n-1}^p + b^p c_{n-1}^l & a_n^p + b^p c_n^l \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ & & \cdots & & \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \tag{57}$$

Since there is no direct way to find the common P satisfying for all possible l and p, trial and error need to be used.

Fuzzy Autoregressive Moving Average Model: In general, the output of a system can be described with a function or a mapping of the plant input-output history. For a single-input single-output (SISO) discrete-time system, the mapping can be written in the form of a nonlinear auto-regressive moving average (NARMA) as follows:

$$y(k+1) = f(y(k),\ y(k-1),\ \cdots,\ u(k),\ u(k-1), \cdots) \tag{58}$$

where $y(k)$ and $u(k)$ are respectively the output and input variables at the k-th time step.
The objective of the control problem is to find a control input sequence which will drive the system to an arbitrary reference set point y_{ref}. Rearranging (58) for control purpose, the value of the input u at the k-th step that is required to yield the reference output y_{ref} can be written as follows:

$$u(k) = g(y_{ref},\ y(k),\ y(k-1),\ \cdots,\ u(k-1),\ u(k-2),\ \cdots) \tag{59}$$

which is viewed as an inverse mapping of (58).

The proposed controller doesn't use rules pre-constructed by experts, but forms rules with input and output history at every sampling step. The rules generated at every sampling step are stored in a rule base, and updated as experience is accumulated using a self-organizing procedure.

The system (58) yields the last output value $y(k+1)$ when the output and input values, $y(k)$, $y(k-1)$, $y(k-2)$, ···, $u(k)$, $u(k-1)$, $u(k-2)$, ···, are given. This implies that $u(k)$ is the input to be applied when the desired output is y_{ref} as indicated explicitly in (59). Therefore, a FARMA rule with the input and output history is defined as follows:

IF y_{ref} is A_{1i}, y(k) is A_{2i}, y(k-1) is A_{3i}, ···, y(k-n+1) is $A_{(n+1)i}$
AND u(k-1) is B_{1i}, u(k-2) is B_{2i} , ···, u(k-m) is B_{mi},
THEN u(k) is C_i, (for the i-th rule) (60)

where, n, m : number of output and input variables
A_{ij}, B_{ij} : antecedent linguistic values for the i-th rule
C_i : consequent linguistic value for the i-th rule.

The rule (60) is generated at (k+1) time step. Therefore, $y(k+1)$ is given value at (k+1) step. The rule (60) explains that "IF desired y_{ref} is $y(k+1)$ with given input-output history, $y(k)$, $y(k-1)$, $y(k-2)$, ···, $u(k)$, $u(k-1)$, $u(k-2)$, ···, THEN $u(k)$ is the input to be applied".

In a conventional FLC, an expert usually determines the linguistic values A_{ij}, B_{ij}, and C_i by partitioning each universe of discourse, and the formulation of fuzzy logic control rules is achieved on the basis of the expert's experience and knowledge. However, these linguistic values can be determined from the crisp values of the input and output history at every sampling step. Therefore, at the initial stage, the assigned $u(k)$ may not be a good control, but over time, the rule base is updated using the self-organizing procedure, and better controls can be applied [16].

A. 2 Adaptive Fuzzy Controller Design

The motivation behind the fuzzy control is to handle uncertainties or unknown variations in model parameters and structures. Similarly, the basic objective of adaptive control is to control systems in the presence of these uncertainties. Therefore, it is natural to combine the two and design *adaptive fuzzy control* [18,19]. Fig. 7 shows the basic configuration of an adaptive fuzzy control system. The reference model is used to specify the ideal response that the controlled system should follow. The plant is assumed to contain unknown parameters. The fuzzy controller is constructed from fuzzy systems whose parameters θ are adjustable. The adaptation law adjusts the parameters θ on-line such that the plant output $y(t)$ tracks the reference model output $y_m(t)$.

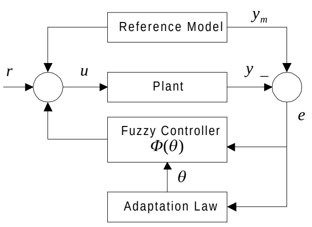


Fig. 7. Adaptive fuzzy control system.

The main advantages of adaptive fuzzy control systems are: (I) better performance is usually achieved because the adaptive fuzzy controller can adjust itself to the changing environment, and (ii) less information about the plant is required because the adaptation law can help to learn the dynamics of the plant during real-time operation. The main disadvantages, on the other hand, are: (i) the resulting control system is more difficult to analyze because it is not only nonlinear but also time-varying, and (ii) implementation is more costly.

Although adaptive fuzzy control and conventional adaptive control are similar in principles and mathematical tools, they differ in the sense that: (i) the fuzzy controller has a special nonlinear structure that is universal for different plants, whereas the structure of a conventional adaptive controller changes from plant to plant, and (ii) human knowledge about the plant dynamics and control strategies can be incorporated into adaptive fuzzy controllers, while such knowledge is not considered in conventional adaptive control systems, which is the main advantage of adaptive fuzzy control over conventional adaptive control.

Human knowledge about a control system can be classified into categories: plant knowledge and control knowledge. Depending upon the human knowledge used and the structure of the fuzzy controller, adaptive fuzzy controller is classified into the following three types:

- Indirect adaptive fuzzy control: The fuzzy controller comprises a number of fuzzy systems constructed from the plant knowledge.
- Direct adaptive fuzzy control: The fuzzy controller is a single fuzzy system constructed from the control knowledge.
- Combined indirect/direct fuzzy control: The fuzzy controller is a weighted average of the indirect and direct adaptive fuzzy controllers.

<u>Indirect Adaptive Fuzzy Controller:</u>

Consider the nonlinear system

$$x^{(n)} = f(x, \dot{x}, \ldots, x^{(n-1)}) + g(x, \dot{x}, \ldots, x^{(n-1)})u \quad (61)$$

where $x \in R$ is the output y, $u \in R$ is the control, $\mathbf{x} = (x, \dot{x}, \ldots, x^{(n-1)})^T$ is the state vector, and f and g are unknown nonlinear functions with $g > 0$ assumed. This type of system can be linearized with nonlinear feedback and a stable linear controller con be designed [11].

Since the functions $f(\mathbf{x})$ and $g(\mathbf{x})$ are unknown, the fuzzy system describes their input-output behavior:

$$IF\ x_1\ is\ F_1^r\ and \cdots and\ x_n\ is\ F_n^r, THEN\ f(\mathbf{x})\ is\ C^r \quad (62)$$

$$IF\ x_1\ is\ G_1^r\ and \cdots and\ x_n\ is\ G_n^r, THEN\ g(\mathbf{x})\ is\ D^r \quad (63)$$

If the nonlinear functions $f(\mathbf{x})$ and $g(\mathbf{x})$ are known, then the feedback linearization controller for the system (61) is given by

$$u^* = \frac{1}{g(\mathbf{x})}[-f(\mathbf{x}) + y_m^{(n)} + \mathbf{k}^T \mathbf{e}] \quad (64)$$

where $e = y_m - y$, $\mathbf{e} = (e, \dot{e}, \ldots, e^{(n-1)})^T$ and $\mathbf{k} = (k_n, \ldots, k_1)^T \in R^n$ is such that all roots of the polynomial $s^n + k_1 s^{n-1} + \cdots + k_n$ are in the left-half complex plane.

The system (61) with control (64) is then rewritten as

$$e^{(n)} + k_1 e^{(n-1)} + \cdots + k_n e = 0 \tag{65}$$

which, because of the choice of **k,** implies $e(t) \to 0$ as $t \to \infty$, i.e., the plant output y converges to the ideal output $y_{m.}$ asymptotically.

Since $f(\mathbf{x})$ and $g(\mathbf{x})$ are unknown, the ideal controller (64) cannot be implemented. However, the fuzzy IF-THEN rules (62)-(63) give estimates $\hat{f}(\mathbf{x}) = \hat{f}(\mathbf{x}|\theta_f)$ and $\hat{g}(\mathbf{x}) = \hat{g}(\mathbf{x}|\theta_g)$, where $\theta_f \in R^{M_f}$ and $\theta_g \in R^{M_g}$ are unknown parameter vectors in $\hat{f}(\mathbf{x})$ and $\hat{g}(\mathbf{x})$, respectively. Thus, the fuzzy controller becomes

$$u = u_I = \frac{1}{\hat{g}(\mathbf{x}|\theta_g)}[-\hat{f}(\mathbf{x}|\theta_f) + y_m^{(n)} + \mathbf{k}^T\mathbf{e}] \tag{66}$$

Typically, the unknown parameters are the centers of the output fuzzy sets C^r and D^r in the rules (62) and (63), respectively. Using the product inference, singleton fuzzifier, and center average defuzzifier, and following the similar procedure leading to (11)-(14), the estimates are in the form:

$$\hat{f}(\mathbf{x}|\theta_f) = \theta_f^T \xi(\mathbf{x}) \tag{67}$$

$$\hat{g}(\mathbf{x}|\theta_g) = \theta_g^T \eta(\mathbf{x}) \tag{68}$$

where $\xi(\mathbf{x})$ and $\eta(\mathbf{x})$ are the fuzzy basis function defined in (12) for fuzzy sets F_i^r and G_i^r, respectively, and θ_f^T and θ_g^T are vectors of the centers of the output fuzzy sets C^r and D^r in the rules (62) and (63), respectively.

Next step is to adjust the parameter vectors θ_f^T and θ_g^T such that the tracking error **e** and the parameter errors $\theta_f - \theta_f^*$ and $\theta_g - \theta_g^*$ are minimized. The Lyapunov synthesis approach defines the following Lyapunov function:

$$\begin{aligned} V = \frac{1}{2}\mathbf{e}^T P\mathbf{e} + \frac{1}{2\gamma_1}(\theta_f - \theta_f^*)^T(\theta_f - \theta_f^*) \\ + \frac{1}{2\gamma_2}(\theta_g - \theta_g^*)^T(\theta_g - \theta_g^*) \end{aligned} \tag{69}$$

where γ_1 and γ_2 are constants and P is a positive matrix satisfying the Lyapunov equation (41).

An adaptation law which minimizes the Lyapunov function is given by [18]

$$\dot{\theta}_f = -\gamma_1 \mathbf{e}^T P\mathbf{b}\xi(\mathbf{x}) \tag{70}$$

$$\dot{\theta}_g = -\gamma_2 \mathbf{e}^T P\mathbf{b}\eta(\mathbf{x})u_I \tag{71}$$

where **b** is defined in (39) with g = 1.

<u>Direct Adaptive Fuzzy Controller:</u>

Consider the nonlinear system

$$x^{(n)} = f(x, \dot{x}, \ldots, x^{(n-1)}) + g(x, \dot{x}, \ldots, x^{(n-1)})u \tag{72}$$

where $x \in R$ is the output y, $u \in R$ is the control, $\mathbf{x} = (x, \dot{x}, \ldots, x^{(n-1)})^T$ is the state vector, and f and g are unknown functions as before. For simplicity, assume that $g = b$, an unknown positive constant. The control objective remains the same as in the indirect adaptive fuzzy control, i.e., design a feedback controller $u = u(\mathbf{x}|\theta)$ based on fuzzy systems and adaptation law for adjusting the parameter vector θ, such that the plant output y follows the ideal output y_m as close as possible. The main difference lies in the assumption about the human knowledge. Instead of knowing the plant knowledge (62) and (63), we are provided with some control knowledge, i.e., the following IF-THEN rules that represent human control actions:

$$IF\ x_1\ is\ P_1^r\ and \cdots and\ x_n\ is\ P_n^r,\ THEN\ u\ is\ Q^r \tag{73}$$

where P_i^r and Q^r are fuzzy sets in R, and r = 1, 2, ...,L_u.

Using the product inference, singleton fuzzifier, and center average defuzzifier, and following the similar procedure leading to (11)-(14), the fuzzy controller is in the form:

$$u_D(\mathbf{x}|\theta) = \theta^T \xi(\mathbf{x}) \tag{74}$$

where $\xi(\mathbf{x})$ and $\eta(\mathbf{x})$ are the fuzzy basis function defined in (12) for fuzzy sets P_i^r, and θ^T is the vector of the centers of the output fuzzy sets Q^r in the rule (73).

Next step is to adjust the parameter vector θ^T such that the tracking error **e** and the parameter error $\theta - \theta^*$ is minimized. The Lyapunov synthesis approach defines the following Lyapunov function:

$$V = \frac{1}{2}\mathbf{e}^T P\mathbf{e} + \frac{b}{2\gamma}(\theta - \theta^*)^T(\theta - \theta_f^*) \tag{75}$$

where γ is a positive constant and P is a positive matrix satisfying the Lyapunov equation (41).

An adaptation law which minimizes the Lyapunov function is given by [18]

$$\dot{\theta} = -\gamma \mathbf{e}^T p_n \xi(\mathbf{x}) \tag{76}$$

where p_n is the last column of P.

Combined Direct/Indirect Adaptive Fuzzy Controller

This adaptive fuzzy controller incorporate both types of linguistic information, plant knowledge and control knowledge. Consider the system (72) with b = 1, for simplicity. Assume that the following information is available:

- Information 1: The plant f in (72) is represented by an approximate model $\hat{f}$.
- Information 2: The modeling error $\tilde{f} = f - \hat{f}$ is given by the fuzzy IF-THEN rules:

$$IF\ x_1\ is\ S_1^r\ and \cdots and\ x_n\ is\ S_n^r, THEN\ \tilde{f}\ is\ E^r \tag{77}$$

where S_i^r and E^r are fuzzy sets in R and r = 1, 2, ..., L_e.

- Information 3: Control actions are given by the fuzzy IF-THEN rules (73).

From (64), if $f(\mathbf{x})$ is known, then the optimal control is

$$u^* = -f(\mathbf{x}) + y_m^{(n)} + \mathbf{k}^T \mathbf{e} \tag{78}$$

to guarantee $y(t) \to y_m(t)$. However, the best estimate of $f(\mathbf{x})$ based on Informaions 1 and 2 is

$$\hat{f}(\mathbf{x}) + \tilde{f}(\mathbf{x}|\theta_I) \tag{79}$$

Thus, the controller based upon Informations 1 and 2 is

$$u_{12} = -\hat{f}(\mathbf{x}) - \tilde{f}(\mathbf{x}|\theta_I) - f(\mathbf{x}) + y_m^{(n)} + \mathbf{k}^T \mathbf{e} \tag{80}$$

The fuzzy controller based upon Information 3 is, from (74),

$$u_3 = u_D(\mathbf{x}|\theta_D) \tag{81}$$

Therefore, the combined fuzzy controller is

$$u = \alpha u_{12} + (1-\alpha) u_3 \tag{82}$$

where $\alpha \in [0,1]$ is a weighting factor.

The fuzzy systems $\tilde{f}(\mathbf{x}|\theta_I)$ and $u_D(\mathbf{x}|\theta_D)$ are respectively designed following the same steps in the indirect and direct fuzzy controller design:

$$\tilde{f}(\mathbf{x}|\theta_I) = \theta_I^T \xi(\mathbf{x}) \tag{83}$$

$$u_D(\mathbf{x}|\theta_D) = \theta_D^T \eta(\mathbf{x}) \tag{84}$$

Next step is to adjust the parameter vectors θ_I^T and θ_D^T such that the tracking error $\mathbf{e}$ and the parameter errors $\theta_I - \theta_I^*$ and $\theta_D - \theta_D^*$ are minimized. The Lyapunov synthesis approach defines the following Lyapunov function:

$$\begin{aligned} V = \frac{1}{2}\mathbf{e}^T P \mathbf{e} + \frac{\alpha}{2\gamma_1}(\theta_I - \theta_I^*)^T(\theta_I - \theta_I^*) \\ + \frac{1-\alpha}{2\gamma_2}(\theta_D - \theta_D^*)^T(\theta_D - \theta_D^*) \end{aligned} \tag{85}$$

where γ_1 and γ_2 are constants and P is a positive matrix satisfying the Lyapunov equation (41).

An adaptation law which minimizes the Lyapunov function is given by [18]

$$\dot{\theta}_I = -\gamma_1 \mathbf{e}^T P \mathbf{b} \xi(\mathbf{x}) \tag{86}$$

$$\dot{\theta}_D = -\gamma_2 \mathbf{e}^T P \mathbf{b} \eta(\mathbf{x}) \tag{87}$$

B. Tuning Controller Performance

The design of an observer and optimal controller is in general based on an assumed linear model that is an approximate representation of an otherwise nonlinear plant. Moreover, the controller takes precise measurements of plant variables and generates a precise control variable. An alternative to this model-based controller design is the fuzzy logic control, which neither relies on an accurate description of the plant, nor on the precise measurements. Fuzzy logic controllers are generally based on experts' understanding of the plant rather than any mathematical model. Another approach is to design a controller based on the knowledge obtained of the system from repeated simulation conducted on a mathematical model. In either case, the rule base of the fuzzy logic controller has to be fine-tuned or calibrated using trial and error in order to obtain the desired performance. Therefore, an Automatic Tuning Method (ATM) is developed to tune the fuzzy logic controller's critical parameters to achieve a desirable response of the plant [3,20].

B.1 Automatic Tuning of Fuzzy Logic Controller

Fuzzy Logic Controller:

The fuzzy logic is based on intuition and experience, and can be regarded as a set of heuristic decision rules or "rules of thumb." One of the most interesting applications of fuzzy logic is the development of fuzzy logic controller. A fuzzy logic controller consists of :

1) A rule base which contains a number of control rules.
2) A database which defines the membership functions of the linguistic terms used in the rule base.

3) An inference mechanism based on the control rules.
4) A fuzzification unit to map real inputs from sensors into the fuzzy terms.
5) A defuzzification unit to map fuzzy outputs of the inference mechanism to real numbers.

A fuzzy logic controller uses a set of control rules and an inference mechanism to determine the control action for a given process state. The control rules are fuzzy expressions that relate the fuzzy process variables (controller inputs) to the fuzzy controller outputs. The inference mechanism evaluates the rule base to find the appropriate control action.

A fuzzy control action consists of situation and action pairs. Conditional rules expressed in *IF* and *THEN* statements are generally used. In order to tune the rule base, the fuzzy controller consists of a number of rules in the form:

$$IF\ f(x_1 \text{ is } A_1,\dots,x_k \text{ is } A_k)\,,\ THEN\ y = g(x_1,\dots,x_k)\,, \quad (89)$$

where x_i and *y* are respective variables of the premise and the consequent, A_i are fuzzy sets with membership functions representing a fuzzy subspace in which the above *IF-THEN* rule can be applied, f is a logical function connecting propositions in the premise, and *g* is a function that implies the value of *y* when $x_1,\dots,x_k$ satisfy the premise.

The consequence (the outputs, or drive) used here are parameterized functions of the input variables. To apply rules like this to fuzzy algorithms for process control, the variables of the premise and the consequent are defined as the following:

Error (*E*) = process output - set point
Error change (*DE*) = current error - last error
Controller output = input applied to process.

The domain of a variable, *E* or *DE*, is partitioned into fuzzy sets, A_i, *i* = 1, 2, $\cdots$. Every fuzzy set A_i is associated with a name that represents qualitative statements, e.g., for *i* = 1, 2, $\cdots$, 5, A_1 = large negative (*LN*), A_2 = small negative (*SN*), A_3 = zero (*ZE*), A_4 = small positive (*SP*), and A_5 = large positive (*LP*). An example of a rule, where the consequent of the rule is a parameterized function of the input variables, is:

IF error (*E*) is large negative (*i* = 1) and the change in error (*DE*) is small negative (*j* = 2),
THEN the output is

$$u_{12} = c_{12}^0 + c_{12}^1 E + c_{12}^2 DE\,, \quad (90)$$

where the subscripts represent $Rule_{12}$, and the parameters c_{12}^k, *k* = 0, 1, and 2, need to be determined. In general the parameters for $Rule_{ij}$, for all *i* and *j*, are determined by the Automatic Tuning Method (ATM) using the input and output data from the experiment [3].

<u>Automatic Tuning Method:</u>

Most existing fuzzy logic controllers are designed without using any mathematical model of a plant. The construction procedures are generally based on the experts' understanding of the process. Therefore, the rule base of a fuzzy logic controller must be adjusted through trial and error to obtain the desired performance. In order to tune the controller, the fuzzy logic controller uses parameterized output functions (90) as the consequent to rules. These parameters permit the use of numerical algorithms to modify the output of the controller.

The consequent of each rule of the controller has the form

$$u_{ij}(k) = c^0 + c_{ij}^1 E(k) + c_{ij}^2 DE(k)\,, \quad (91)$$

where c^0 is known steady-state controller output, and c_{ij}^1 and c_{ij}^2 are the unknown parameters. To find these unknowns, the Kalman filter approach is taken because the Kalman filter estimates are the optimal mean-squared error estimates. Also, in this recursive filter there is no need to store past measurements for the purpose of computing present estimates. In order to apply the Kalman filtering, the unknown parameters c_{ij}^l are viewed as state variables, the premise variables *E*(*k*) and *DE*(*k*) as time-varying system coefficients, and the $u_{ij}(k)$ as the system output variables. Then the dynamics of c_{ij}^l can be modeled simply as a stochastic system in discrete-time:

System Model:

$$\begin{bmatrix} c_{ij}^1(k) \\ c_{ij}^2(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} c_{ij}^1(k-1) \\ c_{ij}^2(k-1) \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} w_{(k-1)}\,, \quad w_k \approx N(0,\infty)\,. \quad (92)$$

Measurement Model:

$$u_{ij} = \begin{bmatrix} E(k) & DE(k) \end{bmatrix} \begin{bmatrix} c_{ij}^1(k) \\ c_{ij}^2(k) \end{bmatrix} + v_k + c^0\,,\ v_k \approx N(0,0)\,, R_k^{-1} = \infty\,, \quad (93)$$

where w_k and v_k are process and measurement noise, respectively, with normal distribution. In this formulation, the process noise is assumed to be completely unknown and the measurement model is assumed to have zero measurement noise. The parameters are unknown constants and therefore their changes at steady-state are zero. Also, the variations of the two parameters are uncorrelated. From these initial assumptions for the system model, the Kalman filtering problem can be easily solved to give the steady-state solution for the parameters c_{ij}^l.

B.2 Self-Organizing Fuzzy Controller Design

The FARMA rule defined in Section 1 is generated at every sampling time. Each rule can be represented as a point in the (n+m+1)-dimensional rule space, i.e., $(x_{1i}, x_{2i}, \dots, x_{(n+m+1)i})$. To update the rule base, the following performance index is defined:

$$J = |y_r(k+1) - y(k+1)|, \tag{94}$$

where y(k+1) is the real plant output, and $y_r(k+1)$ is the reference output. Therefore, at the (k+1)-th step, the performance index J is calculated with the real plant output y(k+1) resulting from the k-th step control.

The fuzzy rule space is partitioned into a finite number of domains and only one rule, i.e., a point, is stored in each domain. If there are two rules in a given domain, the selection of a rule is based on J. That is, if there is a new rule which has the output closer to the reference output in a given domain, the old rule is replaced by the new one. The self-organization of the rule base, in other words "learning" of the object system, is performed at each sampling time, Fig. 8 [16,17].

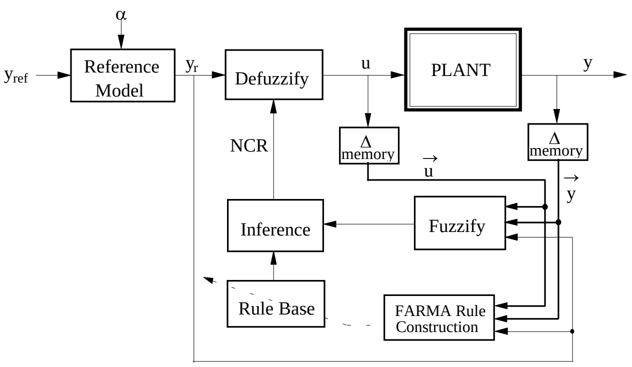


Fig. 8. The FARMA control system architecture.

C. References

[1] L.-X. Wang, *A Course in Fuzzy Systems and Control*. Prentice-Hall PTR, Upper Saddle River, NJ, 1997.

[2] Takagi and M. Sugeno, “Fuzzy identification of systems and its applications to modeling and control,” *IEEE Trans. on Systems, Man, and Cybern.*, vol. 15, no. 1, pp. 116-132, 1985.

[3] P. Ramaswamy, R. M. Edwards, and K. Y., Lee, “An Automated Tuning Method of a Fuzzy Logic Controller for Nuclear Reactors,” *IEEE Transactions on Nuclear Science*, 40, pp. 1253-1262, August 1993.

[4] M. Vidyasagar, *Nonlinear Systems Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1993.

[5] A. E. Bryson and Y. C. Ho, *Applied Optimal Control*, Hemisphere, NY, 1975.

[6] V. I. Utkin, *Sliding Modes and Their Application to Variable Structure Systems*, MIR Publishers, Moscow, 1978.

[7] J. E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice-Hall, Englewood Cliffs, NJ, 1991.

[8] R. Palm, “Sliding mode fuzzy control,” *Proc. 1st IEEE Inter. Conf. on Fuzzy Systems*, pp. 519-526, San Diego, 1992.

[9] D. Driankov, H. Hellendoorn, and M. Reinfrank, *An Introduction to Fuzzy Control*, Springer-Verlag, Berlin, 1993.

[10] L.-X. Wang, “A supervisory controller for fuzzy control systems that guarantees stability,” *IEEE Trans. on Automatic Control*, 39, no. 9, pp. 1845-1848.

[11] A. Isidori, *Nonlinear Control Systems*, Springer-Verlag, 1989.

[12] Z. Y. Zhao, M. Tomizuka, and S. Isaka, “Fuzzy gain scheduling of PID controllers,” *IEEE Trans. on Systems, Man, and Cybern.*, 23, no. 5, pp. 1392-1398.

[13] T. Takagi and M. Sugeno, “Fuzzy identification of systems and its applications to modeling and control,” *IEEE Trans. on Man, and Cybern.*, 15,1, pp. 116-132, 1985.

[14] M. Sugeno and G. T. Kang, “Structure identification of fuzzy model,” *Fuzzy Sets and Systems*, 28, no. 1, pp. 15-33, 1988.

[15] Tanaka and M. Sugeno, “Stability analysis and design of fuzzy control systems,” *Fuzzy Sets and Systems*, 45, no. 2, pp. 135-156, 1992.

[16] Y. M. Park, U. C. Moon, and K. Y. Lee, “Self-Organization of Fuzzy Logic Controller for Dynamic Systems Using a Fuzzy Auto-Regressive Moving Average (FARMA) Model,” *IEEE Trans. on Fuzzy Systems*, 3, pp. 75-82, February 1995.

[17] Y. M. Park, U. C. Moon, and K. Y. Lee, “A self-organizing power system stabilizer using fuzzy auto-regressive moving average (FARMA) model,” *IEEE Trans. on Energy conversion*, 11, 2, pp. 442-448, 1996.

[18] L.-X. Wang, *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1994.

[19] T. A. Johansen, “Fuzzy model based control: stability, robustness, and performance issues,” *IEEE Trans. on Fuzzy Systems*, 2, pp. 221-234, 1994.

[20] P. Ramaswamy, M. Riese, R. M. Edwards, and K. Y. Lee, “Two Approaches for Automating the Tuning Process of Fuzzy Logic Controllers,” *Proc. of the IEEE 32nd Conference on Decision and Control*, pp. 1753-1758, San Antonio, TX, Dec. 15-17, 1993.

Chapter 5 Expert System Applications

Abstract - Fuzzy logic allows a convenient way to incorporate the knowledge of human experts into the expert systems using qualitative and natural language-like expressions. Recent advances in the field of fuzzy systems and a number of successful real-world applications in power systems show that logic can be efficiently applied to deal with imprecision, ambiguity and probabilistic information in input data. Fuzzy logic based systems with their capability to deal with incomplete information, imprecision, and incorporation of qualitative knowledge have shown great potential for application in electric load forecasting.

A. Expert Systems

The major use of artificial intelligence today is in expert systems, AI programs that act as intelligent advisors or consultants. Drawing on stored knowledge in a specific domain, an inexperienced user applies inferencing capability to tap the knowledge base. As a result, almost anyone can solve problems and make decisions in a subject area nearly as well as an expert.

It is not easy to give a precise definition of an expert system, because the concept of expert system itself is changing as technological advances in computer systems take place and new tasks are incorporated into the old ones. In simple words, it can be defined as a computer program that models the reasoning and action processes of a human expert in a given problem area. Expert systems, like human experts, attempt to reason within specific knowledge domains.

An expert system permits the knowledge and experience of one or more experts to be captured and stored in a computer. This knowledge can then be used by anyone requiring it. The purpose of an expert system is not to replace the experts, but simply to make their knowledge and experience more widely available. Typically there are more problems to solve than there are experts available to handle them. The expert system permits others to increase their productivity, improve the quality of their decisions, or simply to solve problems when an expert is not available.

Valuable knowledge is a major resource and it often lies with only a few experts. It is important to capture that knowledge so others can use it. Experts retire, get sick, move on to other fields, and otherwise become unavailable. Thus the knowledge is lost. Books can capture some knowledge, but they leave the problem of application up to the reader. Expert systems provide a direct means of applying expertise.

An expert system has three main components: a *knowledge base*, an *inference engine*, and a *man-machine interface.* The knowledge base is the set of rules describing the domain knowledge for use in problem solving. The prime element of the man-machine interface is a working memory which serves to store information from the user of the system and the intermediate results of knowledge processing. The inference engine uses the domain knowledge together with the acquired information about the problem to reason and provide expert solution.

A Working Definition

An expert system is an artificial intelligence (AI) program incorporating a knowledge base and an inferencing system. It is a highly specialized piece of software that attempts to duplicate the function of an expert in some field of expertise. The program acts as an intelligent consultant or advisor in the domain of interest, capturing the knowledge of one or more experts. Non-experts can then tap the expert system to answer questions, solve problems, and make decisions in the domain.

The expert system is a fresh new, innovative way to capture and package knowledge. Its strength lies in its ability to be put to practical use when an expert is not available. Expert systems make knowledge more widely available and help overcome the age-old problem of translating knowledge into practical, useful results. It is one more way that technology is helping us get a hand on the oversupply of information. All AI software is knowledge-based as it contains useful facts, data, and relationships that are applied to a problem.

Expert systems, however, are a special type of knowledge-based system, they contain heuristic knowledge. Heuristics are primarily from real world experience, not from textbooks. It is knowledge that directly from those people -the experts - who have worked for years within the domain. It is knowledge derived from learning by doing. It is perhaps the most useful kind of knowledge, specifically related to everyday problems. It has been said that knowledge is power. Certainly there is truth in that but in a more practical sense, knowledge becomes power only when it is applied. The bottom line in any field of endeavor is RESULTS, some positive benefit or outcome. Expert systems are one more way to achieve results faster and easier.

A.1 Desirable Expert System Features

Expert systems are far more useful if they have some additional features. These include an explanation facility, ease of modification, transportability, and adaptive learning ability. Let's take a look at each of these key features.

Explanation Facility

Expert systems are very impersonal and get right to the point. Many first time users are surprised at how quickly the expert system comes up with a recommendation, conclusion, or selection. The result is usually stated concisely, and sometimes very curtly, using rule clauses. A natural language interface will help improve this situation, but that's not the main problem. A more important issue is that often users have difficulty in "buying" the output decision. They question it or perhaps don't believe it. Users frequently want to know how the expert system arrived at that answer. Most of the better expert systems have a means for explaining their conclusion. Typically, this takes the form of showing the rules involved in the decision and the sequence in which they were fired. All of the information is retained in the data base for that purpose. When users want to know the expert system's line of reasoning, they can read the rules and follow the logic themselves. Some rule formats permit the inclusion of an explanation statement that justifies or elaborates on the need for or importance of the rule.

The explanation facility is important because it helps the user feel comfortable with the outcome. Sometimes the outcome is a surprise or somewhat different than expected. It is difficult for an individual to follow the advice of the expert in these cases. However, once the expert system explains itself, the user better understands the decision and feels more at ease in making a decision based upon it.

Ease of Modification

As indicated earlier, the integrity of the knowledge base depends upon how accurate and up to date it is. In domains where rapid changes take place, it is important that some means be provided for quickly and easily incorporating this knowledge. When the expert system was developed using one of the newer development tools, it is usually a simple matter to modify the knowledge base by writing new rules, modifying existing rules, or removing rules. The better systems have special software subsystems which allow these changes to be made without difficulty. If the system has been programmed in LISP or Prolog, changes are much more difficult to make. In examining or evaluating an expert system, this feature should be considered seriously in context of the modification.

Transportability

The wider the availability of an expert system the more useful the system will be. An expert system is usually designed to operate on one particular type of computer, and this is usually dictated by the software development tools used to create the expert system. If the expert system will operate on only one type of computer, its potential exposure is reduced. The more different types of computers for which the expert system is available, the more widely the expertise can be used. If possible, when the expert system is to be developed, it should be done in such a way that it is readily transportable to different types of machines. This may mean choosing a programming language or software development tool that is available on more than one target machine.

Adaptive Learning Ability

This is an advanced feature of some expert systems that allows them to learn their own use or experience. As the expert system is being operated, the engine will draw conclusions that can, in fact, produce new knowledge. New functions stored temporarily in the data base, but in some systems they can lead to the development of a new rule which can be stored in the knowledge base and used again in the problem. The more the system is used, the more it learns about the domain and more valuable it becomes.

The term learning as applied to expert systems refers to the process of the expert system new things by adding additional rules or modifying existing rules. On the other hand, if the system incorporates the ability to learn it becomes a much more powerful and effective problem solver. Today few expert systems have this capability, but it is a feature that is sure to be further developed into future systems.

A.2 Suitable Application Areas for Expert Systems

Expert systems are best suited for problems with limited domains and well-defined expertise. Application areas involving common sense and analogical reasoning do not lend themselves well to expert system development. The suitability of expert system-based approaches can be determined by taking into consideration some criterion based on general experience in this field. Expert systems are found to be suitable for those problems for which the solution steps are not clearly defined. The action taken depends not only on the present values of data but on the outcome of previous decisions, historical data, past experience and trends.

In power systems, many promising applications have been reported in the broad fields of system control, alarm processing and fault diagnosis, system monitoring, decision support, system analysis and planning. An excellent review of the popular application areas can be found in [1].

A.3 Expert System Applications

Expert systems are ideal when it is necessary for an individual to select the best alternative from a long list of choices. Based on the criteria supplied to it, the expert system can choose the best option. For example, there are expert systems that will help you select one of the many places to invest your money based on your own financial condition, goals, and personality traits.

An expert system can be created to help an individual troubleshoot and repair a complex piece of equipment. The various troubles and symptoms can be given to an expert system which then identifies the problem and suggests courses of action for repair. Expert systems also can be used to aid in diagnosing medical cases. Symptoms and test results can be given to the expert system which then searches its knowledge base in an attempt to match these input conditions with a particular malady or disease. This results in a conclusion about the illness and some possible suggestions on how to treat it. Such an expert system can greatly aid a doctor in diagnosing an illness and prescribing treatment. It does not replace doctors, but helps them confirm their own decisions and may provide alternative conclusions.

Expert systems perform financial analysis. Some expert systems evaluate stocks and recommend buy, sell, or hold positions. Other expert systems can be used in tax planning and budgeting. Expert systems have been used to help locate oil and mineral deposits or to configure complex computer systems and recommend a specific policy in a variety of insurance applications. Expert systems also have been used to locate oil spills and provide speedy critical advice to commanders in battlefield situations. The variety of potential applications is enormous. If one or more experts exist in the domain of interest, and the knowledge can be codified and represented in symbolic form, then an expert system can be created.

In power systems, many promising applications have been reported in the broad field of system control, alarm processing and fault diagnosis, system monitoring, decision support, system analysis and planning. An excellent review of the popular application areas can be found in [1].

Table 1 shows the main categories of applications suitable for expert systems. If the problem to be solved falls into one of these categories, it is a candidate for expert systems solution. This is not to imply that an expert system is the only answer. There may very well be a more conventional algorithmic program that will do the job. In any case, assuming the problem is one of these types, an expert system should most certainly be considered as an alternative. Now let's take a look at each category in more detail.

Table 1. Generic Expert System Categories

Control - intelligent automation
Debugging- renovation corrections to faults
Design-development products to specification
Diagnosis- estimated defects
Instruction-optimized computer instruction
Interpretation-clarification of situations
Planning-developing goal-oriented scheme
Prediction-intelligent guessing of outcome
Repair-automatic diagnosis, debugging, planning and fixing

B. Reasoning with Uncertainty in Rule Based Expert Systems

One of the important feature in expert systems is their ability to deal with incorrect or uncertain information. There will be times when an expert system, in gathering initial inputs, will ask you a question for which you do not have the answer. In such a case, you simply say that you do not know. Expert systems are designed to deal with cases such as this. Because you may not have a particular fact, the search process will undoubtedly take a different path. It may take longer to come up with an answer, but the expert system will give you an answer.

Traditional algorithmic software simply cannot deal with incomplete information. If you leave out a piece of data, you may not receive an answer at all. If the data is incorrect**,** the answer will be incorrect. This is where artificial intelligence programs, particularly expert systems, are particularly useful. When the inputs are ambiguous or completely missing, the program may still find a solution to your problem. The system may qualify that solution, but at least it is an answer that can in many cases be put to practical use. This is consistent with expert level problem solving where one rarely has all the facts before making a decision. Our common sense or knowledge of the problem tells us what is important to know and what is less important. Experts almost always work with incomplete or questionable information, but that it doesn't prevent them from solving the problem.

Thus, increasingly in the design of expert systems, there has been a focus on methods of obtaining approximate solutions to a problem when there is no clear conclusion from the given data. Logically, as expert system problems become more complex, the difficulty of reaching a conclusion with complete certainty increases, so in some cases, there must be a method of handing uncertainty. In [2,3], researchers report that a classical expert system gave incorrect results due to the sharpness of the boundaries created by the if-then rules of the system; however, once a method for dealing with uncertainty (in these two cases fuzzy set theory) was used, the expert system reached the desired conclusions.

The successful performance of expert systems relies heavily on human expert knowledge derived from domain experts based on their experience. The other forms of knowledge include causal knowledge and information from case-studies, databases, etc. Knowledge is typically expressed in the form of high level rules. The expert knowledge takes the form of heuristics, procedural rules and strategies in nature.

It inherently contains vagueness and imprecision because an expert is not able to explicitly express their knowledge. The process of acquiring knowledge is also quite imprecise, because the expert is usually not aware of all the tools used in the reasoning process. The knowledge that one reasons with may itself contain uncertainty. Uncertain data and incomplete information are other sources of uncertainty in expert systems.

Uncertainty in rule based expert systems occurs in two forms. The first form is linguistic uncertainty which occurs if an antecedent contains vague statements such as the level is high" or "the value is near 20". The other form of uncertainty, called evidential uncertainty, occurs if the relationship between an observation and a conclusion is not entirely certain. This type of uncertainty is most commonly handled using conditional probability which indicates the likelihood that a particular observation leads to a specific conclusion. The study of making decisions under either of these types of uncertainty will be referred to as plausible or approximate reasoning in this work. Several methods of dealing with uncertainty in expert systems have been proposed, including

- Subjective probability
- Certainty factors
- Fuzzy measures
- Fuzzy set theory

The first three methods are generally used to handle evidential uncertainty, while the last method, fuzzy set theory is used to incorporate linguistic uncertainty. These methods of reasoning with uncertainty will be discussed in the following sections. For a comprehensive list of methods used in reasoning with uncertainty including a discussion about their application. see [4, pp. 1307-1322].

As expert assessments of the indicators of the problem may be imprecise, fuzzy sets may be used for determining the degree to which a rule from the expert system applies to the data that is analyzed. When applying a method of reasoning with uncertainty to a rule based expert system, there must be a method of combining or propagating uncertainty between rules. A method of propagating uncertainty for the method of reasoning with uncertainty will be discussed in the next section.

B.1 Subjective Probability and Statistics

One method of dealing with uncertainty is to use conventional statistics and probability. For example, with the use of statistics, sufficient data may be available to compute mean (average), median, and standard deviation. These new figures derived from original data provide additional knowledge which will help in making a decision. Recall that probability is simply a ratio the number of times that a particular action will occur for a given number of attempts. It is really a ratio as shown below:

> *P(x)* =Number of occurrence of an event /Total number of events that take place

The probability of x occurring, stated as *P(x)*, is the ratio of the number of times x occurs to the total number of events that take place. For example, in rolling a standard die, the probability is one-sixth that any one of numbers 1 through 6 will come up. This may also be expressed as a fraction, .16667, or as a percentage, 16.67%. In many knowledge representation cases, the probability for a certain condition or action may be known or can be estimated. For the probability of a certain event taking place is 70%, then it may initiate some action if the probability is equal or greater than 70%. If the probability is less than 70%, then perhaps an action may not be taken. For example, the production rule below uses the probability:

> IF the stone is clear, without color
> THEN it is diamond (probability 60%)

An example will illustrate this. Suppose we ask ten engineers whether they can program in the BASIC language. Out of the ten, three say they can. We can use these figures to compute the probability:

$$P(BASIC) = \frac{3}{10} = 0.3$$

What this says is that the probability of an engineer being able to program in BASIC is .3. We can also express this as a percentage by simply multiplying the probability by 100. We say that the probability of engineers being able to program in BASIC is 30%. Probability figures like this can be used to determine rule strength if they fit the problem.

Multiple probability values will occur in many systems. For example, a rule may have three parts to its antecedent, each with a probability value. The overall probability of the rule then becomes the product of the individual probabilities, if the parts of the antecedent are independent of one another. In a three part antecedent, the probabilities may be .9, .7 and .65. The overall probability is:

$$P = (.9)(.7)(.65) = .4095$$

The combined probability is about 41%. But this is true only if the individual parts of the antecedent do not affect or depend on one another.

Sometimes one rule references another. Here the individual rule probabilities can propagate from one to another. There is a need to evaluate the total probability of a sequence of rules or a path through the search tree to determine if a specific rule fires. Or you may be able to use the combined probability to predict the best path through the search tree. In other words, the probabilities become the "costs" of the individual arcs in the tree.

There are numerous methods of computing combined probabilities. If the rules are independent, a simple product can be used as described before. However, most events and rules are dependent upon one another. In that case, a special procedure called Bayes' Rule or Theorem can compute the probability of event A occurring given that event B has already occurred. This is expressed as *P(A|B)*. Bayes' Theorem is:

$$P(A|B)=\frac{P(B|A)P(A)}{P(B|A)P(A)+P(B|\sim A)P(\sim A)}$$

We won't attempt to explain this rule here as it is doubtful that you would ever need to program it. But you should know that many expert systems use Bayes' Theorem instead of certainty factors to deal with uncertainty. Several major expert system development tools use Bayesian probability.

B.2 Measures of Belief and Disbelief

Measures of belief arose disbelief arose from the desire that evidence should incrementally increase the belief or disbelief in a hypothesis. The formal definition of the measure of belief was based on the idea that if a prior probability, *P(h)* is defined, then the maximum amount of belief that can be added to *P(h)* from a new piece of evidence is 1 - *P(h)*. If a piece of evidence confirms *P(h\e)*, then this would amount to adding *P(h\e)* - *P(h)* to the previous belief, so the belief in *h* has been increased by

$$\Delta P(h) = \frac{P(h|e) - P(h)}{1 - P(h)}$$

The measure of increased disbelief can be defined similarly. Now with this idea, let the measure of increased belief (*MB*) given some evidence *e* about a hypothesis *h* be defined as *MB[h,e]*$\rightarrow$[0,l] with

$$MB(h,e)=\begin{cases} 1 & \text{if } P(h)=1 \\ \max\left\{0,\dfrac{P(h|e)-P(h)}{1-P(h)}\right\} & \text{otherwise} \end{cases}$$

and let the measure of increased disbelief (MD) be defined as *MD[h, e]*$\rightarrow$ [0, 1] with

$$MD(h,e)=\begin{cases} 1 & \text{if } P(h)=0 \\ \max\left\{0,\dfrac{P(h)-P(h|e)}{1-P(h)}\right\} & \text{otherwise} \end{cases}$$

Note that when evidence e is assigned to a hypothesis *h*, only one of the *MD* or *MB* functions will be greater than zero so that a single piece of evidence cannot be used as both a measure of the confirmation and negation of a hypothesis. As the measures of belief and disbelief were used in the design of expert system, it was found that a representation of the uncertainty in terms of a single measure would be more convenient in making comparisons of different hypothesis.

B.3 Certainty Factors

As you saw earlier, there are several methods of dealing with uncertain information. In rule based expert systems, numerical factors indicating the truth or probability of a premise or conclusion are used as a measure for uncertainty. These numerical factors are known as certainty factors (CF) and probability. In this section we want to take a look at these measures of uncertainty to be sure that you understand their use in rule based expert systems.

In a high percentage of expert system rules, there will be no ambiguity or uncertainty. We will know with confidence whether or not a particular premise or conclusion is true or false. If the information is not known at all, then the rule requesting it will not fire. In cases where there is the possibility that the information is not known, special rules can be created to deal with this problem. The rule might state that if a particular piece of information is not available, then a certain action will be initiated.

Still, there are many cases where the information is known but we have less than 100% confidence in its truthfulness. Just as weather forecasters use a number to predict the likelihood of rain, so can a confidence number be used with production rules. Weather forecasters may say that there is a 70% probability of rain. They are saying that they don't know for sure whether or not it is going to rain. On the other hand, they have enough information to be able to say that 70% of the time under similar circumstances it does rain. While a certainty or confidence factor is not really a probability, it is a number that helps you to represent the uncertainty. A certainty. factor is simply a measure of the confidence you have that a particular fact or rule is true or not true. It is usually a number between 0 and 1 where zero indicates no confidence and 1 means full or complete confidence. You will also hear certainty factors called confidence factors or rule strength.

Certainty factors are used with both the premise (IF) and conclusion (THEN) portions of a rule. The two examples given below show how confidence factors are used.

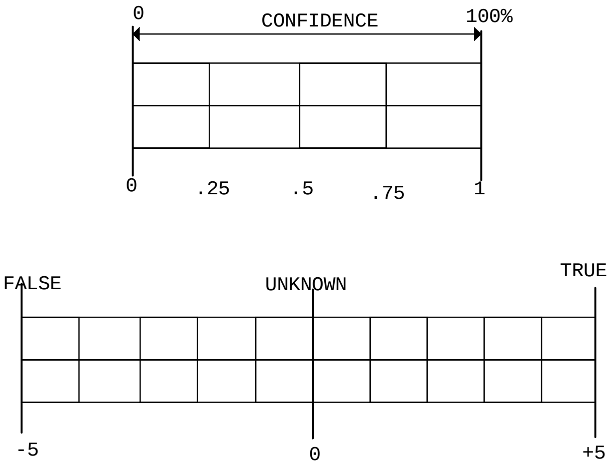


Figure 1. Confidence and certainty factor scales

IF	the patient has hayfever, CF = .6
THEN	prescribe an antihistamine

IF	the patient is sneezing
AND	has a runny nose
AND	has watery eyes
THEN	the patient has a cold, CF=.5

Fig. 1 shows several ways to use certainty factors. As you can see, the scale is up to the programmer. In example A, a scale of 0 to 1 is used where 1 = absolute certainty; that is, 100% truthfulness or validity of the premise or conclusion of a rule. The 0, of course, indicates absolute uncertainty or falsity. Intermediate values have varying degrees of truthfulness or uncertainty. You could also use a scale of 0 to 10 or 0 to 100 with the same result. The + and - scale shown in example B in Fig. 1 is another approach. A + 5 indicates absolute certainty while a - 5 indicates 100% contradiction. The 0 in the center of the scale indicates unknown. You could also use a -1/ 0 / +1 scale as well.

Determining whether a particular rule is to fire requires the inference engine to look at the confidence factor and evaluate it. For example, if you are using the 0 to 1 scale, you might want the rule to fire if the confidence factor is above a certain threshold level, say a 0.2. In Fig. 1B, you may assign a threshold of + 1 or - 1 depending upon the circumstances as the minimum acceptable level for determining whether something is true or false. Other levels may be set depending upon the problem.

In rules with compound premise clauses connected by AND or OR, each clause may have its own CF. For such situations, there must be a way to compute a composite CF for the rule. This is done by using the minimum CF of all clauses connected by AND or the maximum CF of all clauses connected by OR. Some examples will illustrate this.

Rule 1:
IF X (.4)
AND Y (.75)
THEN Z
Composite CF = .4

Rule 2:
IF D (.3)
AND E (.8)
THEN F
Composite CF = .8

If each rule in a reasoning chain has a CF, each will, of course, affect the other. The outcome has to be decided based upon some composite evaluation. One way this is done is with a special formula.

$$CF = CF(X) + CF(Y) - CF(X)*CF(Y)$$

This says that the CF for rule X is added to the CF for rule Y and from that is subtracted the product of the CFs for rules X and Y. Below are two rules to illustrate the point.

Rule 3:
IF P
AND Q
THEN R (.65)

Rule 4:
IF R
THEN S (.2)

The composite CF then is:

$$.65 + -2 - (.65)\ (.2) = .85 - .13 = .72$$

Of course, there will usually be more than two rules in a chain. The formula above can be used by taking the composite CF of two rules and combining it with the CF of a third rule. That new composite CF is then combined with a fourth, and so on.

B.4 Fuzzy Logic

Another method of dealing with imprecise or uncertain knowledge is to use fuzzy *logic*. Fuzzy logic is a system conceived by Zadeh for dealing in inexact or unreliable information. In this method, an attempt is made to assign numerical ranges with a possibility value between zero and one to concepts such as height, beauty, age, and other elements with values that are hard to pin down. It allows you to work with ambiguous or fuzzy quantities such as large or small, or data that is subject to interpretation.

For example, how tall is tall? Are you tall if you are 5 foot 7 inches? Is tall over 6 feet or over 6 feet 3 inches? Is short

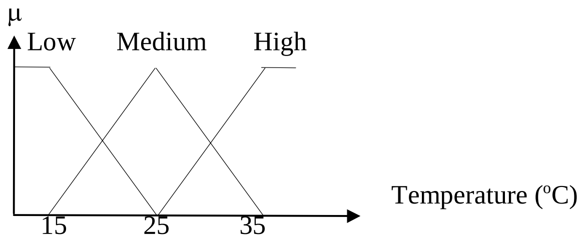


Figure 2: Fuzzy sets for representation of uncertainty

less than 5 foot 5 inches or what? Fuzzy logic gives you a way of expressing this kind of approximate information. Tall might be expressed as some value of X where X is between 5'10' and 6'2' with a possibility of .8. A "0" possibility means that X is not in the range given while "1" means that X is between the values given. Values between 0 and 1 mean some degree of possibility that X is in the given range. Once you are able to express such imprecise knowledge, you can use it more reliably in the reasoning process. Fuzzy logic is not as widely used in expert systems as confidence factors and probability because it is more complex and difficult to implement. And often it does not offer any advantages over the simpler systems. But, it is an alternative of growing importance as AI expands into new areas of application.

Fuzzy set-based techniques can provide an excellent framework for systematically representing the imprecision inherent in an expert's knowledge. Using the following example to illustrate,

> IF the temperature is high (0.8)
> AND system is operating in heavy load period (0.9)
> THEN system is highly stressed

the parameters in premise and consequent (temperature, load period, system stress) can be represented using simple fuzzy membership functions. In this case, the fuzzy information is contained in the terms high and heavy. An example is shown in Fig. 2 where temperature is divided into three linguistic classes each [*low (L), medium (M), high (H)*]. If the forecast temperature is 32°C, it's membership in class [high] will be 0.7.

The load period can also be represented using similar fuzzy variables. If the hour of the day falls during heavy load period with a membership grade of 0.87, the membership of the consequent can be obtained as the minimum of the two, i.e., 0.7.

Although, there are situations where membership grades and probabilities can take on similar values, they are not the same. One distinguishing factor between probability and fuzzy membership grades is that the summation of probabilities on a finite universal set must equal to 1.

The main drawback of nonfuzzy methods in dealing with uncertainty is their handling of linguistic terms. Fuzzy set theory provides a natural framework for dealing with linguistic terms used by experts. Imprecision in numeric data can be easily dealt with by expressing it as a fuzzy number. Fuzzy sets can be conveniently incorporated in expert systems to better deal with uncertainty and imprecision.

C. Example Application - Fault Diagnosis

In the past few years, great emphasis has been put in applying the expert systems for transmission system fault diagnosis. However, very few papers deal with the unavoidable uncertainties that occur during operation involving the fault location and other available information. This example shows a method using fuzzy sets to cope with such uncertainties.

C.1 Problem Statement

To reduce the outage time and enhance service reliability, it is essential for dispatchers to locate fault sections in a power system as soon as possible. Currently, heuristic rules from dispatchers' past experiences are extensively used in fault diagnosis. The important role of such experience has motivated extensive recent work [5-11] on the application of expert system in this field. A few papers have described and dealt with uncertainties involving the fault location and other information available [12-15]. These uncertainties occur due to failures of protective relays and breakers, errors of local acquisition and transmission, and inaccurate occurrence time, etc. An effective approach is thus necessary to deal with uncertainties in these expert systems.
Fault diagnosis in electric power system is a facet operation. Every signal and step contain some uncertainties, which can be modeled by membership functions. Fuzzy set theory is used to determine the most likely fault sections in the approach presented here. Membership functions of the possible fault sections are the most important factors in the inference procedures and decision making. In this example, the membership function of a hypothesis is used to describe the extent to which the available information and the system knowledge match the hypothesis. They are manipulated during inference based on rules concerning fault sections.

C.2 Structure of the Fault Diagnosis System

The fuzzy expert system structure is shown in Fig. 3. Its database contains the power system topology, and the status of all breakers and protective relays after the fault.

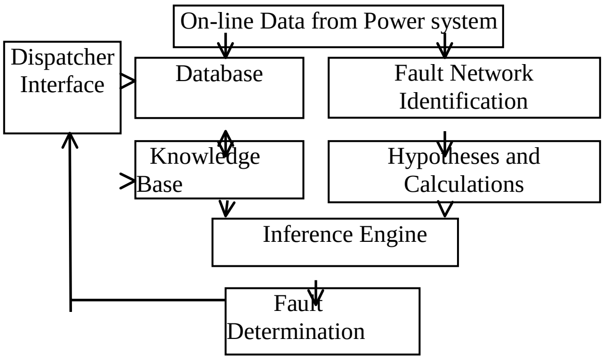


Fig. 3 Fuzzy Expert system Structure

The knowledge base of the fuzzy expert system contains all the data of the protection system. The information is based on known statistics of protection performance used in the system. If these data are not available when a fault occurs, the fuzzy expert system asks the dispatcher to provide them and then saves them in the database for future use. Models for estimation of possible faults, and heuristic rules about the relay characteristics for actual fault determination are also included here.

C.3 Island Identification

When a fault occurs in a power system, the relays corresponding to the fault sections should trip the circuit breakers to isolate the fault sections from being extended. Thus the power system is separated into several parts named subnetworks after the operation of protective relays and circuit breakers. Generally, only a few subsections are formed from the faults. Since the fault sections are confined to these subnetworks, The magnitude of the problem can be reduced greatly.

An expert system is developed to identify the island by using the real-time information of circuit breakers and adopting the real-time network topology determination method [17]. The framework of this efficient method is described as follows:

- Initializing the network: The expert system identifies the power system pre-fault status as the normal operation state by using the real-time network topology determination method [17]. When a fault occurs, the power system status would be changed by the operation of relays and circuit breakers.
- Healthy subnetwork identification: The next step is to identify the network topology of the healthy part of the post-fault power system by using the real-time network topology determination method [18]. The healthy subnetwork is called set $S_{healthy}$.
- Island identification: By comparing the initial network topology with the healthy subnetwork topology, the differences between them are identified as the island. This subnetwork is called S_{island}.

This method was proven in a case study that consists of 43 substations, 523 sections, 412 circuit breakers, 107 busbar, 23 three-winding transformers and 77 transmission. The simulating results are quite satisfactory [17]. The required processing time to identify the island is less than 2 seconds in a 486 micro-computer in all the simulated cases.

C.4 Fault Section Identification

When a fault occurs, the change in breaker status activates the fuzzy expert system. It then classifies the breakers into two sets: no-trip status set and tripped status set. According to the procedures described in section 3, a fault hypothesis F_i is formed as follows:

$$F_i = F_i(CB) \cup F_i(RL) = \{(C_i, \mu_{P_{fault}}(C_i) | C_i \in S_{island}\} \tag{1}$$

$$P_{fault} = \{F_i\} \tag{2}$$

$$F_i(CB) = \{C_i, \mu^{CB}_{P_{fault}}(C_i)\} \tag{3}$$

$$F_i(RL) = \{C_i, \mu^{RL}_{P_{fault}}(C_i)\} \tag{4}$$

where C_i is one of the possible fault sections being considered; P_{fault} is the fuzzy set which contain all the possible fault sections and their membership functions; $F_i(CB)$ is the fuzzy subset by considering only the tripped circuit breakers; $F_i(RL)$ is the fuzzy subset by considering only the operated relays; $\mu^{CB}_{P_{fault}}(C_i)$ is the membership function that C_i belongs to the fuzzy set P_{fault} by considering only the tripped circuit breakers; $\mu^{RL}_{P_{fault}}(C_i)$ is the membership function that C_i belongs to fuzzy set P_{fault} by considering only the operated relays.

Table 2 Formation of Island for Case Study 1

Fault section subnetwork	Tripped circuit breakers	CB12-A, CB12-B, CB12-D, CB2D, CB21-C, CB23-A, CB32-B, CB42-A, CB54-A, CB54-B
identification	Island	F12-A, F12-B, F12-D, F2AB, F2BC, F2CD, F2DA, F23-B, F24-A, F24-B, F4AA, F4BB, F45-A, F45-B

Table 3Membership Grades of Case Study 1

Possible fault section	F2DA	F12-A	F24-A	F45-A	F12-D	F23-B	F12-B
Membership function	0.950	0.489	0.489	0.489	0.489	0.489	0.477

Possible fault section	F45-B	F4AA	F2CD	F2AB	F2BC	F4BB	F24-B
Membership function	0.476	0.348	0.285	0.282	0.0356	0.0356	0.0238

Table 4Diagnosis Results of Case Study 1.1

Estimated Fault section	Tripped circuit breakers	Maloperated breakers
Busbar F2DA	CB12-A, CB12-B, CB12-D, CB2D, CB21-C, CB23-A, CB32-B, CB42-A, CB54-A, CB54-B	CB2A

The following rules are used to determine the overall grade of the results.

Rule 1: If (first stage protection has operated)
then (ignore the signals in second and third stage protections)

Rule 2: If (first and second protection have isolated the suspected fault section)
then (ignore the signals of third stage protection)

Rule 3: If (all three stage protections have not isolated the suspected fault section)
then (no fault at this section)

During decision making, the most likely fault sections are determined by comparing the above membership grades for each possible fault section using either or both of the following methods [12] :

(1) **Maximum Selection:** The most likely fault section is the one with the highest membership grade $\mu_{P_{fault}}(C_i)$.

(2) **α-level selection**: The α-level set includes all fault sections with a membership grade $\mu_{P_{fault}}(C_i)$ greater than the qualifying value of α.

According to the methods described above, two additional rules are formed for decision making:

Rule 4: (Maximum selection):
If (section M has the greatest membership function compared with all the other possible fault sections)
then (select M as the fault section)

Rule 5: (α-level selection):
If (the membership grade of a fault section is greater than the constant α)
then (add this section to the fault section set)

<u>Case Study</u>

Fig. 4 shows the test network. Both substations SS1 and SS2 have double buses with 4 bus-ties, and 4 transmission lines between them. In this example, the primary bus protection relay (type PR) “PR2DA” has operated, but “CB2A” has failed to trip. Back-up relays (type BR) “BR12-A”, “BR12-B”, “BR12-D”, BR32-B”, and “BR42-A” have operated, but “BR42-B” has failed to detect the fault. Therefore, “BR54-A” and “BR54-B” have operated, making “CB54-A” and CB54-B” to trip accordingly.

Based on the tripped circuit breaker status, the island is as shown in Table 2.

Using Equations 1-4, the membership grade of all possible fault sections are calculated and shown in Table 3.

Finally, the rule of maximum selection described above is used and only the section F2DA is selected as the fault section because the membership grade of this section is much higher than those of other sections. The diagnosis results are correct and are shown in Table 4.

D. Conclusion

Generally, a conventional rule-based expert system for bulk power system needs several hundreds of rules. It is time-consuming in inference procedures to search for suitable rules during inferencing. On the other hand, fuzzy set based expert systems tend to be much faster compared to traditional rule-based expert systems for most of the rules are replaced by the calculation of the membership functions of the applicable rules . Only a few rules or functions are used in the inference engine.

The fuzzy set approach for uncertainty processing in expert systems offers many advantages to compared other approaches to deal with uncertainty.

- *Small memory space and computer time*: The knowledge base is very small because there are only a few rules needed during inference. The computation time is therefore also small.
- *Small number of rules*: With properly designed linguistic variables and level of granularity, only a few fuzzy rules are needed for each situation.
- *Flexibility of the system*: Membership functions representing the parameters can be changed dynamically according to the situation. It is also possible to develop a self-learning module that modifies the grades of membership automatically according to changing situations.

E. References

[1] T. S. Dillon and M. A. Laughton, *Expert System Applications in Power Systems,* Prentice Hall International Series in Power Systems Computation.

[2] N. Vadiee and M. Jamshidi, "The Promising Future of Fuzzy Logic," *IEEE Expert*, August 1994, pg. 37.

[3] R. Levi, M. Rivers and K. Hickey, "An Intelligent Instrument for High Voltage Equipment Insulation Evaluation," *Proceedings of the 1994 International Conference on Intelligent System Application to Power Systems*, France, Sept. 1994, pp. 471-476.

[4] S.C. Shapiro, ed., *Encyclopedia of Artificial Intelligence*, Wiley, New York, 1992.

[5] Yuan-Yih Hsu, F. C. Lu, Y. Chien, et al., "An expert system for locating distribution system faults," *IEEE Trans. on PWRD*, Vol. 5, No. 2, pp366-371.

[6] M. Kezunovic, C. W. Fromen, et al., "An expert system for transmission substation event analysis," IEEE Trans. on Power Delivery, Vol.8, No. 4, Oct. 1993, pp1942-1949.

[7] J.R. McDonald, G.M.Burt D.J. Young, "Alarm processing and fault diagnosis using knowledge based systems for transmission and distribution network control," *IEEE Trans. on Power Systems*, Vol.7, No.3, Aug. 1992, pp1292-1298.

[8] Takafumi Kimura, Sinya Nishimatsu, et al., "Development of an expert system for estimating fault section in control center based on protective system simulation," *IEEE Trans. on Power Delivery*, Vol.7, No.1, Jan., 1992, pp167-172.

[9] C.A. Protopapas, A.V. Machias, et al., "An expert system for substation fault diagnosis and alarm processing," *IEEE Trans. on Power Delivery*, Vol.6, No.2, April 1991, pp648-655.

[10] Benoit Valiquette, Germano Lambert Torres, "An expert system based diagnosis and advisor tool for teaching power system operation emergency control strategies," *IEEE Trans. on Power Systems*, Vol.6, No.3, Aug. 1991, pp1315-1322.

[11] Chihiro Fukui, Junzo Kawakami, "An expert system for fault section estimation using information from protective relays and circuit breakers," *IEEE Trans. on Power Delivery*, Vol. PWRD-1, No.4, Oct. 1986.

[12] C.S Chang, J.M. Chen et al., "Power system fault diagnosis using fuzzy sets for uncertainties processing," accepted for publication in ISAP'96, Orlando, USA.

[13] C.S. Chang, J.M. Chen, A. C. Liew, "An expert system approach for fault diagnosis considering uncertainties," *Proceedings of International Conference on Intelligent Manufactureing'95 (ICIM'95)*, Wuhan, China, June, 1995.

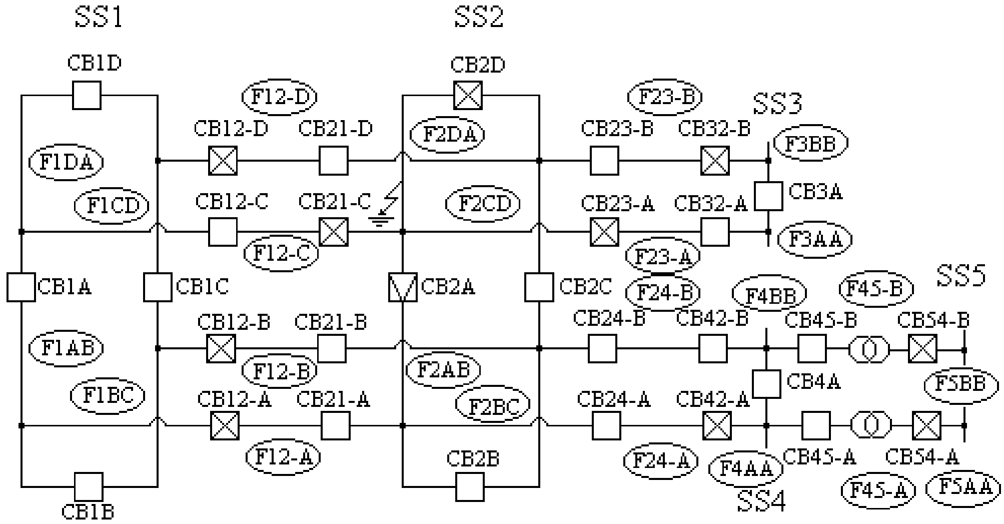
[14] Cho H.J., Park J. K., Lee H. J., "A fuzzy expert system for fault diagnosis of power systems," *Proceedings of ISAP'94, Montpellier*, France, Sept., 1994, pp217-222.

[15] P. Jarventausta, P. Verho, J. Partanen, "Using fuzzy sets to model the uncertainty in the fault location process of distribution networks," *IEEE Trans. on Power Delivery*, Vol.9, No. 2, April 1994, pp954-960.

[16] Zhu Yongli, Y.H. Yang, B. W. Hogg, W.Q. Zhang, S. Gao, "An expert system for power systems fault analysis," *IEEE Trans. on Power Systems*, Vol. 9, No. 1, Feb., 1994, pp503-509.

[17] Fushan Wen and Zhenxiang Han, "Optimal fault section estimation in power systems using a genetic algorithm," accepted for *Journal of Electric Power Systems Research*.

[18] M. Prais, "A topology processor that tracks network modifications over time," *IEEE-PWRS*, Vol.3, No. 3, 1989, pp992-998.



Notes: Closed circuit breakers

Tripped Circuit Breakers

Relay operated normally, circuit breaker failed to trip

Fig. 4. Operating Relays and Tripped Circuit Breakers

Chapter 6 Optimization Techniques I

A. Session Overview

Power system planning and operating often face the mutli-objective optimization situations, which is a challenging problem. Conventionally, solutions of a multi-objective optimization problem are given as Pareto solutions, of which consist of many solution points. The decision maker needs to decide or select one specific solution out of the many solutions by considering various factors relating to the problem at hand. In general, the decision will be made based on his/her preference, experience, or subjective judgment.

Three kinds of information are involved in the decision process : *Goals*, *Constraints*, and *Alternatives*. *Goals* are what we want to achieve out of the decision process. Constraints/criteria are limiting factors that are needed to consider before deciding how to achieve the goals. *Alternatives* are the available decision outcome of the problem under consideration. The typical multi-objective decision problem, which basically a decision process, involves the selection of one alternative, a_i, from a universe of n alternatives $A = \{a_1, a_2, \cdots, a_n\}$ given a set of r objectives $O = \{o_1, o_2, \cdots, o_r\}$ with corresponding weighting factors/constraints constraints $B = \{b_1, b_2, \cdots, b_r\}$ that are important to the decision making. Each alternative will be evaluated on how well it satisfies each objective.

Generally, heuristic intuition, expert knowledge and experience, and linguistic descriptions are very important in the decision making process. Since some information only can be described imprecisely and some others only as quality, the decision making environment is fuzzy. The conventional multi-objective decision making scheme does not capture imprecise information and quality data in an effective manner.

Fuzzy logic technology is a rich field with a large amount of theory and operations developed. Chapter 2 has briefly described the techniques in fuzzy logic, including membership functions, fuzzy rules, fuzzification, defuzzification. This chapter will focus on how to use fuzzy logic to optimize multi-objective decision making for power system problems.

B. Brief Overivew of Yeger's Multi-Objective Decision Making and Linear Fuzzy Linear Programming

This section gives an overview of some fundamental concept of multi-objective optimization decision making and some technical background of Yager's fuzzy multi-objective decision making technique and fuzzy linear programming technique.

B.1 Yager's Fuzzy Logic Multi-Objective Decision Making Scheme

In order to use Yager's Fuzzy Logic Multi-Objective Decision Making Scheme to solve some power engineering problems, there are several issues needed to be addressed. To name a few:

1. How to implement linguistic descriptions of the problem at hand?
2. How to aggregate the available information for decision making?
3. How to infer the final decision based on the aggregated information?

Three popular fuzzy logic concept and techniques can be used to answer the issues posted. *Membership functions* will be used to convert the input values to the linguistic descriptions and membership values to fuzzy descriptive forms. The fuzzy inference such as using min operation to aggregate input information based on the fuzzy rules.

Centroid rule, which is the popular method to perform defuzzification in fuzzy logic [11], can be used to evaluate the fuzzy outputs into an preference value for each alternatives for later decision making approaches used to infer the final decision making based on the preference value obtained from the Centroid rules and the weightings of objectives. The overall decision process is shown in Fig. 1.

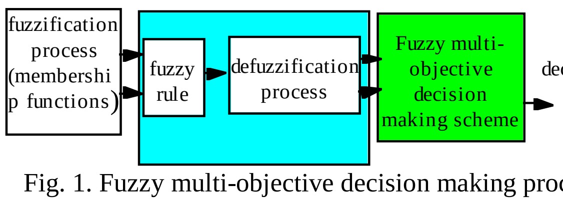


Fig. 1. Fuzzy multi-objective decision making process

In 1981, Yager proposed an approach for decision making that required only *ordinal* information on the ranking of preferences and importance weights. This process naturally requires subjective information from the decision maker concerning the importance of each objective. Based on the multi-objective decision making formulation described previously, the decision measure for a particular alternative, *a*, can be replaced with a classical implication of the form,

$$M(O_i(a), b_i) = b_i \rightarrow O_i(a) = \bar{b}_i \vee O_i(a) . \quad (1)$$

where $\bar{b}_i = 1 - b_i$ and $\vee$ is the *max* operator, i.e.,

$$\bar{b}_i \vee O_i(a) = \max\{\bar{b}_i, O_i(a)\} . \quad (2)$$

The implication preserves the linear ordering required of the preference set, and at the same time relates the two quantities in a logical way where negation is also accommodated. Justification of the implication as an appropriate measure can be developed using an intuitive argument [10]. A reasonable decision model will be the joint intersection of *r* decision measures,

$$J = \bigcap_{i=1}^{r} \left(\bar{b}_i \cup O_i\right), \quad (3)$$

and the optimum solution, a^*, is the alternative that maximizes *J*. If we define

$$C_i = \bar{b}_i \cup O_i , \quad (4)$$

hence

$$\mu_{C_i}(a) = \max\left[\mu_{\bar{b}_i}(a), \mu_{O_i}(a)\right], \quad (5)$$

then the optimum solution, expressed in membership form, is given by :

$$\mu_D(a^*) = \max_{a \in A} \left[\min\left\{\mu_{C_1}(a), \mu_{C_2}(a), \cdots, \mu_{C_r}(a)\right\}\right] \quad (6)$$

Yager's decision making requires users to rank the group of goals and the group of constraints along a comparative scale of importance from 0 to 1 (create fuzzy membership functions for each input and output. Then measure each of the user's alternatives against each of the goals and constraints and rank them from 0 to 1 (another membership function concept).

The preference weighting factors, $B = \{b_1, b_2, \cdots b_r\}$, will be assigned to each of the objectives to quantify the decision maker's feelings about the influence that each objective would have on the chosen alternative. They are used to convert the multiple objectives into an overall decision function in some plausible way. The negation of the preference weighting $\bar{b}_j$ acts as a barrier such that all distinctions less than that barrier is disregarded while those distinctions above the barrier is kept. The more important is the objective, the lower is the barrier, and thus the more level of distinction there are. Later sections will apply Yager's multi-objective on a Power Distribution Spatial Load Forecasting problem.

B.2 Fuzzy Linear Programming

In real world decision problems, objectives and constraints are seldom rigid or crisp but rather vague in the degree of attainment. Fuzzy mathematical programming has been developed significantly in recent years to solve a class of multi-objective optimization problems with ambiguous or fuzzy constraints as well as objectives. Fuzzy linear programming is an effective method of making coordination among many conflicting or trade-off objectives. The coordination will be done through the shape of membership functions assigned to objectives and also to constraints. If the goal of a certain objective is not thought of much, this must be adjusted by redefining the associated membership function. Furthermore, it is advantageous to treat future demand prediction as fuzzy number. The conventional multi-objective linear programming problem with *k* objectives may be formulated as follows:

$$\left.\begin{array}{ll} \text{minimize} & \mathbf{z}(\mathbf{x}) = \mathbf{C}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x} \le \mathbf{b} \\ & \mathbf{x} \ge \mathbf{0} \end{array}\right\} \quad (7)$$

where $\mathbf{z}^T(\mathbf{x}) = [z_1(\mathbf{x}),\ z_2(\mathbf{x}), \cdots,\ z_k(\mathbf{x})]$, $\mathbf{C} = [\mathbf{c}_1,\ \mathbf{c}_2, \cdots,\ \mathbf{c}_k]$, $A^T = [\mathbf{a}_1,\ \mathbf{a}_2, \cdots,\ \mathbf{a}_m]$, and $\mathbf{x},\ \mathbf{c}_i,\ \mathbf{a}_i \in \Re^n$.

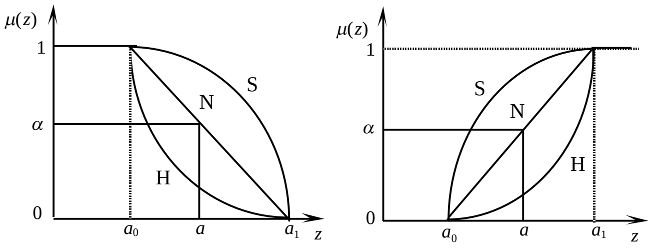
In real world problems, the constraints are mostly given as fuzzy quantities, in other words, by ambiguous or soft constraints. The objectives are rarely need to be minimized or maximized absolutely. However, the objectives should attain values less than some target ranges in the minimization process. The linear programming with fuzzy objectives and constraints may be formulated as:

$$\left.\begin{array}{ll} \text{minimize} & \mathbf{z}(\mathbf{x}) = \mathbf{C}^T\mathbf{x} \lesssim \mathbf{z}^{\circ} \\ \text{subject to} & A\mathbf{x} \lesssim \mathbf{b} \\ & \mathbf{x} \ge \mathbf{0} \end{array}\right\} \quad (8)$$

where, symbol $\lesssim$ denotes fuzzy inequality and is used to express both fuzzy objectives and constraints. For instance, $A\mathbf{x} \lesssim \mathbf{b}$ means the left-hand term is roughly less than the right-hand term.

In fuzzy linear programming, the vagueness of the decision maker in selecting an adequate plan is reflected as the degree of satisfaction through the shapes of membership functions. Therefore, how to set up a membership function may characterizes constraints as well as objectives. Depending on the strictness of objective attainment, we shall assume the three kinds of membership functions as shown in Fig. 2. Here, type S is for the case of low priority, type H is for more stringent case, and type N is linear membership function. In this Figure, relationship between the value of an objective function a and the degree of satisfaction α is shown for the linear membership function.

Type H (Hard) : High priority in an objective
Type S (Soft) : Low priority in an objective
Type N (Normal) : Linear membership function



(a) Minimization (b) Maximization

Fig. 2. Membership function.

Later sections will demonstrate the application of Yager's multi-objective decision technique for Power Distribution Spatial Load Forecasting, demonstrate fuzzy linear programming techniques for Generation Expansion Planning, modified fuzzy linear programming approach to solve Optimal Power Flow problem, and the use Lagrange relaxation techniques on fuzzy linear programming on Scheduling problems.

C. Application of Fuzzy Optimization Methods to Power System Applications

C.1 Application of Yager's Fuzzy Multi-objective Optimization Method for Power Distribution System Spatial Load Forecasting

(by Mo-yuen Chow and JinXiang Zhu, North Carolina State University)

Introduction

Distribution systems aim to provide reliable power to customers in a large geographic areas. In the planning stages, utilities need to plan ahead for anticipated future load growth under different possible scenarios. Based on load forecasts, they will decide whether to built new facilities or upgrade the existing facilities. Their decision can affect the earning or losing millions of dollars for their companies as well as customers' satisfaction and operational reliability. Therefore, decision making tools are very important to make a right decision based on given information. But the correct plan must rely on the accurate load forecasting.

For distribution plan, not only the load magnitude but also its location are to be predicted. The load in distribution level is highly stochastic ('needle peaks') and greatly affected by land usage, weather, and living habits. Apparently, the distribution load forecasting is a high dimensional, stochastic, nonlinear, and time varying problem. It is difficult to identify mathematical models or statistical regression models that have been used successfully in generation and transmission load forecasting.

There are a lot of unforeseen situations may occur and land usage may change through time. For example, the new construction of a highway, the move in of a large industry plant. These external factors can substantially affect the land usage, thus the load growth. The distribution system planners need to aggregate different types of information to predict what might happen in their service areas in the future and plan accordingly. Load studies shows that the land usage dominate the load growth pattern and load shapes because they employ similar type of appliances and have similar needs and schedule [1].

The land usage based spatial load forecasting computer simulation has been proposed and used to aggregate appropriate geographic information to simulate future load growth based on different anticipated scenarios [2]. The increasingly popular, affordable, and accurate Geographic Information Systems (GIS) technology provide an excellent data base platform for spatial load forecasting techniques. The use of GIS can save thousands of man hours for utilities to collect relevant geographic data [3]. Thus spatial load forecasting technology become even more attractive than before both from economical point of view and superior load forecasting accuracy.

There are a few stages for spatial load forecasting, shown in Fig. 3. The spatial information is used to predict the land usage. Each land usage is mapped to a load growth pattern. The land usage and load growth are then calibrated based on different constraints, such as system load growth, budget available, future economy growth of the area, etc.

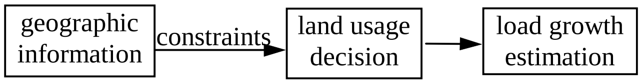


Fig. 3. diagram of land-use based spatial load forecasting.

Spatial Load Forecasting Problem

As discussed in [10], three kinds of information are involved in the decision process : *Goals*, *Constraints*, and *Alternatives*. For the spatial load forecasting problems, we have to identify them before formulation.

Goals are what we want to achieve out of the decision process. In the land usage spatial load forecasting program, distribution engineers want to predict the likelihood of land usage changes in the future due to different influential factors, then estimate the spatial load growth patterns accordingly under different scenarios in order to plan the distribution system ahead of *time*. The land usage goals can be further categorized as:

1. Determine whether the land needs re-development,

2. Determine what land class the site will become if re-development is required, and
3. Determine the corresponding load growth for the land usage.

Constraints/criteria are limiting factors that are needed to consider before deciding how to achieve the goals. The land usage constraints/criteria can be *Land-use Preferences*, *Budget Limitations*, *Geographic Constraints*, etc.

Alternatives in the land usage decision are the available choices for land usage under consideration. For example, the alternatives are different land class usage: Vacant land, Light Residential, ..., Heavy Industry.

<u>Spatial Load Forecasting Description</u>

One of the major process in the land-use based spatial load forecasting process relies on the prediction of future land usage. The choice of land usage belongs to the multi-objective decision evaluation problem based on different factors. The typical multi-objective decision problem, which basically a decision process, involves the selection of one alternative, a_i, from a universe of n alternatives $A = \{a_1, a_2, \cdots, a_n\}$ given a set of r objectives/criteria a_i that are important to the decision making. Each alternative will be evaluated on how well it satisfies each objective.

Distance to highway concept is straight forward [1]. Urban pole concept has been used in city planning and modeling [14]. Among a city or town, site preference may attracted to or repulse from some salient point of geographic interests such as center of district, shopping centers, ball parks. The influence of the center of interest is often presented by the Urban Pole concept [1].

For example, there are three alternatives - residential, commercial, and industrial - are considered for land-use selection in the spatial load forecasting problems [9], that is, n = 3 and $A = \{a_1, a_2, a_3\}$. Suppose two objectives are considered - O_1 distance from highway and O_2 distance from urban pole - then r = 2 and $O = \{O_1, O_2\}$.

<u>Implementation of Land-use Selection</u>

This section presents a land usage based spatial load forecasting prototype demonstration of using fuzzy logic decision making scheme of the land usage determination, from which predicts the future spatial load growth.

<u>Multi-Objective Decision Problem Set-Up</u>

As mentioned in previous sections, the illustration problem is formulated as following :

A 10×10 land grid sites assuming all environmental conditions are the same except the distance to *the* highway, which is under construction, and distance to the urban pole center. There are three alternatives for land usage, such as, residential, commercial, industrial.

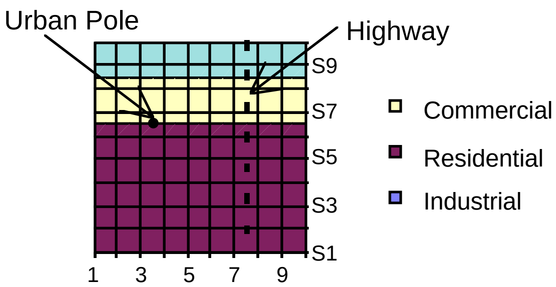


Fig. 4. The illustrative example

The goals of the land usage selection are :

1. To maximize the land value by satisfy the preference of decision makers.
2. To minimize the redevelopment costs.

The inputs for the decision making process is :

1. The distance of the site to the highway.
2. The distance of the site to the urban pole.
3. The original land-use information.
4. The cost of redevelopment from one land-use to another.

<u>Membership Function Set-up</u>

The preferences of the land usage depends on the two external factors: distance to highway, D_h, and distance to urban pole, D_u. The distance to highway and distance to urban pole are described by linguistic variables : *very close* (*V*), *moderately close* (*C*), and *far* (*F*). The membership functions representing these variables are shown in Fig. 5.

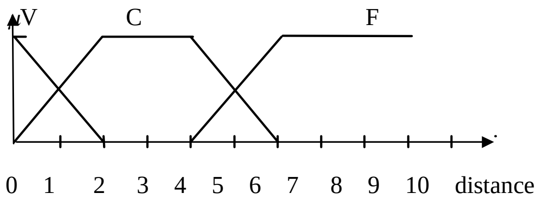


Fig.5. Distance membership functions.

Since the grid size under consideration is 10×10, therefore, the universe of discourse of the input variable is [0, 10]. The preference values are normalized between [0, 1], in which 1 indicates completely *prefer* and 0 indicates completely *against*.

Fig. 6 shows five membership functions to describe the different site preference: *strongly against (SA), moderately against (MA), neutral (NT), moderately prefer (MP), strongly prefer (SP)*. Again, the preference membership functions are normalized between [0,1].

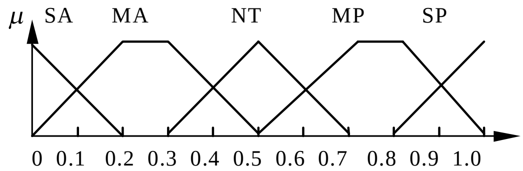


Fig. 6. Preference membership functions.

Fuzzy Rules

The rules for selecting land usage with respect to distance to highway and urban pole can be described by the linguistic variables in the Table 1,2. It means '*If a site is very close to highway, then residential will moderately against*'.

Table 1: Fuzzy rules for the distance to highway

Highway	Residential	Commercial	Industrial
Very Close	MA	SP	SP
Close	SP	NT	SA
Far	MA	MA	SA

Table 2: Fuzzy rules for the distance to urban pole

Urban Pole	Residential	Commercial	Industrial
Very Close	SA	SP	SA
Close	MP	MP	MA
Far	MA	MA	SP

Not only the distance to highway and urban pole, but also redevelopment cost is considered in land-use selection. The redevelopment cost is listed in Table 3. Since cost is to be minimized, the preference of redevelopment is defined as $1-T_{ij}$, where T_{ij} is the cost of redevelopment from *i*th land-use to *j*th land-use.

Table 3: The redevelopment costs

	Resid.	Comm.	Industrial
Residential	0.0	0.2	0.3
Commercial	0.3	0.0	0.4
Industrial	0.3	0.2	0.0

The importance weighting factor of each objective *b* is set differently based on the decision maker's preference. For example, highway: $b_1 = 0.7$, urban pole: $b_2 = 0.6$, cost: $b_3 = 0.5$, that means highway criterion is more important than the urban pole criterion while urban pole objective is more important than the cost criterion.

The centroid rule is applied to defuzzify the preferences to highway and urban pole. These preferences and the preference of redevelopment on cost issue are aggregated by their important weighting factors based on the Yager's approach.

For example, site (S6, 5) is a residential site which is 1.5 miles away from highway and 2.55 miles from urban pole.

- fuzzification to get $\widetilde{D}(d_H) = \{V / 0.25, C / 0.75, F / 0\}$
- apply all applicable rules $\widetilde{P}_R = \{MA / 0.25, SP / 0.75\}$
- defuzzification to get a crisp preference $p_R = 0.8125$

Same procedure will evaluate other two alternatives and get:

$\widetilde{P}_C = \{SP / 0.25, NT / 0.75\}$ $\quad p_C = 0.625$

$\widetilde{P}_I = \{SP / 0.25, SA / 0.75\}$ $\quad p_I = 0.25$

Similarly to the urban pole and cost, the results are listed in Table 4. Each alternative is evaluated based on Equ. (1-6) and the highest rank win. Conclusion: this site is best served by residential.

Table 4. Results for Land-use selection

	weight	Res.	Comm.	Ind.
Highway	0.7	0.8125	0.625	0.25
Urban Pole	0.6	0.75	0.75	0.25
Cost	0.5	1.0	0.8	0.7
Rank		0.75	0.625	0.3

Spatial Load Forecasting

Different loads have their own characteristics and land-use load curves. Reasonable approximations and simplifications have been studied on load growth patterns [1, 15]. These techniques have been used in several power areas such as load modeling, load forecasting, and demand side management.

In this paper, each land use has its own load growth pattern and is described by state-space description in the form of :

$$\dot{S} = aS + b, \tag{9}$$

with appropriate units. The parameters for different land usage used in this paper are listed in Table 5.

Table 5. Load growth parameters used in the illustration.

	R	C	I
a	$\begin{cases} 0.75 \text{ if } S < c \\ -0.75 \text{ if } S \geq c \end{cases}$	$\begin{cases} 0.4 \text{ if } S < c \\ -0.4 \text{ if } S \geq c \end{cases}$	$\begin{cases} 0.6 \text{ if } S < c \\ -0.6 \text{ if } S \geq c \end{cases}$
b	$\begin{cases} 0 \text{ if } S < c \\ 7.5 \text{ if } S \geq c \end{cases}$	$\begin{cases} 0 \text{ if } S < c \\ 6 \text{ if } S \geq c \end{cases}$	$\begin{cases} 0 \text{ if } S < c \\ 12 \text{ if } S \geq c \end{cases}$
c	5	7.5	10
S_0	0.5	0.5	0.5

The load growth pattern described in Eqn. (9) has been shown to be a good approximation for many load growths observed in the past. The parameters *a*, *b*, *c* can be fine tuned to suit the specific problems at hand [1].

Results and Discussion

Land-use selection

The result of land-use redevelopment is shown in the Fig. 7.

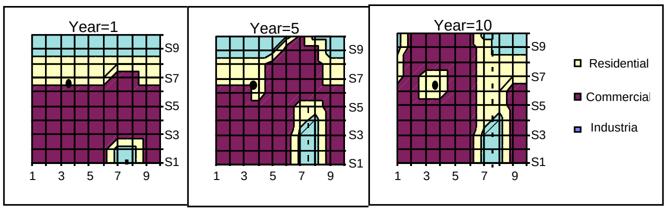


Fig. 7. The final land-use map

Based on the land-use selection rules, the sites around urban pole are the most preferred place for commercial. Therefore some residential sites, which are close to urban pole, are re-developed to commercial sites. On the other hand, some commercial sites and industrial sites which are neither *very close* nor *far* to urban pole nor highway are changed to residential sites. Since industrial sites are *strongly against* to be close to the urban pole but strongly prefer to be close to highways, the sites on the side of highway which is close to urban pole are re-developed to commercial sites while those far away from urban pole are re-developed to industrial sites. These results are consistent with the fuzzy rules and membership functions used.

Once the land-use redevelopment plan has been provided, the spatial load forecasts are easy to obtain. For example, site (S4, 7) will be redeveloped from residential to commercial in third years. After redeveloped it will follow the growth pattern for commercial instead of residential. In Fig. 8, different land-use has different load growth pattern. Site (S1,3) is residential, site (S6,3) is commercial, site (S10,8) is industrial site. Since the land-use redevelopment, the load will change accordingly. Based on the spatial load forecast results, distribution planning software can provide the best feeder design in the future.

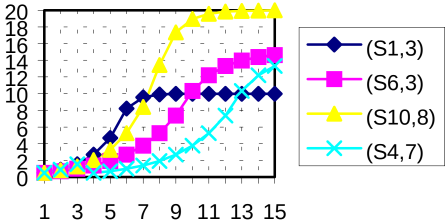


Fig. 8. The electricity consumption.

The advantages of the proposed approach

The fuzzy logic formulation provides a intuitive and easy approach to implement heuristic rules into the spatial load forecasting land-use selection criteria. The fuzzy algorithm is robust even uncertainties employed. In this paper, the decision is made based on the compromise of preference to highway, urban pole, and redevelopment cost. Yager's approach is used to evaluate multi-objective by the importance weighting factors. Another advantage is to easily match the decision maker's expectation, that is, the fuzzy rules and membership functions can be modified to fine tune the results. The fuzzy multi-objective decision making process is robust, easy to fine tune, and easy to maintenance.

C.2 References

[1] H. L. Willis and G. B. Rackliffe, *Introduction to Integrated Resource T&D Planning*: Automated Distribution Division, ABB Power T&D Company Inc., 1994.

[2] H. L. Willis and T. W. Parks, "Fast Algorithms for Small Area Electric Load Forecasting," *IEEE Transactions on Power Systems*, vol. PAS-102, pp. 1111-7, 1983.

[3] ESRI, *Understanding GIS*, 6 ed: ESRI, 1992.

[4] H. L. Willis and J. E. D. Northcote-Green, "Comparison Tests of Fourteen Distribution Load Forecasting Methods," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-103, pp. 1190-1197, 1984.

[5] M.-y. Chow, "Fuzzy Logic Based Control," in *CRC Press Industrial Electronics Handbook*, D. Irwin, Ed., 1995.

[6] M.-y. Chow, "Fuzzy Systems," in *CRC Press Industrial Electronics Handbook*, D. Irwin, Ed.: CRC, 1996.

[7] M.-y. Chow and H. Tram, "Application of Urban Re-Development Methodology in Spatial Load Forecasting," *submitted to IEEE Transactions on Power Systems*, 1996.

[8] M.-y. Chow and H. Tram, "Methodology of Urban Re-Development Considerations in Spatial Load Forecasting," *accepted for publication in IEEE Transactions on Power Systems*, 1996.

[9] M.-y. Chow and H. Tram, "Applications of Fuzzy Logic Technology for Spatial Load Forecasting," *accepted for publication in IEEE Transactions on Power Systems*, 1996.

[10] T. J. Ross, *Fuzzy Logic with Engineering Applications*: McGraw Hill, 1995.

[11] B. Kosko, *Neural Networks and Fuzzy Systems: A Dynamic Systems Approach to Machine Intelligence*. Englewood Cliffs, NJ: Prentice Hall, 1992.

[12] R. Yager, “A New Methodology for Ordinal Multi-objective Decisions Based on Fuzzy Sets,” *Decision Science*, vol. 12, pp. 589-600, 1981.
[13] L. A. Zadeh, “Fuzzy Sets,” *Information and Control*, vol. 8, pp. 338-353, 1965.
[14] H. L. Willis and H. Tram, “Tutorial on Spatial Load Forecasting,” *IEEE*, 1991.
[15] H. L. Willis, R. W. Powell, and H. N. Tram, “Computerized Methods for Analysis of Impact of Demand Side Management on Distribution Systems,” *IEEE Transactions on Power Delivery*, pp. 1236, 1987.

D. Application of Fuzzy Linear Programming with Lagrangian Relaxation Technique to Power Scheduling and Transactions

(Sen Lin and Peter Luh, University of Connecticut)

D.1 Scheduling and Transactions in an Uncertain Environment

Since electric utilities generally have different sets of generators and have to meet their time-varying demand and reserve requirements, they usually have different marginal generation costs. It is often mutually beneficial to buy or sell power if their marginal costs are sufficiently different. The transaction problem, however, is difficult because transactions are coupled with scheduling through demand and reserve requirements, and the scheduling problem itself is “NP hard.” In addition, significant uncertainties are involved, including demand, reserve, and future transaction opportunities. These uncertainties have major impact on the economics of system operation. In view of the increasing competitiveness of the power market, prudent transaction decisions are required to be made in almost real time. A good problem formulation and an effective methodology are needed to manage uncertainties.

In this subsection, the integrated scheduling and transaction problem is formulated as a mixed fuzzy-stochastic optimization problem in subsection b. To solve the problem with manageable complexity, a method based on a synergistic combination of fuzzy optimization, Lagrangian relaxation, and stochastic dynamic programming is developed in subsection c. Testing results based on data from Northeast Utilities presented in subsection d demonstrate that the algorithm is robust, significant savings are obtained, and a good balance is achieved between minimizing costs and hedging against uncertainties.

D.2 Problem Formulation

To simplify the presentation, a power system with *I* thermal units, M future purchase transactions, and N future sale transactions is considered. Hydro and pumped-storage units and given transaction opportunities (having crisp prices and maximum levels) can be handled by following our previous work as reported in [1-4]. They are in fact considered in numerical testing presented in subsection d. The objective is to minimize the total cost, i.e., the fuel and start-up costs plus the expected purchase costs and minus the expected sale revenue, subject to fuzzy demand and reserve requirements and individual unit and transaction constraints. The time unit is one hour, and the planning horizon may vary from one week to ten days. To formulate the scheduling and transaction problem mathematically, the following notation is introduced:

$C_i(p_i(t))$: Cost of thermal unit i at hour t, in dollars;
I: Total number of thermal units;
$P_d(t)$: System demand at hour t, in MW;
$P_r(t)$: System reserve requirement at hour t, in MW;
$p_i(t)$: Power generated by thermal unit i at hour t, in MW;
$r_i(p_i(t))$: Reserve contribution of thermal unit i at generation level $p_i(t)$, in MW;
$S_i(t)$: Start up cost of thermal unit i in \$, a linear function of time since last shut down;
T: Scheduling horizon, in hours.

Object Function. The objective is to minimize the total cost -- the fuel and start-up costs plus the expected purchase costs and minus the expected sale revenues:

$$J \equiv \sum_{i=1}^{I}\left[C_i(p_i(t)) + S_i(t)\right] + \sum_{m=1}^{M} E\left[c_m(t)p_m(t)\right] - \sum_{n=1}^{N} E\left[c_n(t)p_n(t)\right]. \qquad (10)$$

Systemwide Constraints. In view of the inaccurate forecasted demand and uncertain transaction opportunities, the system demand and reserve are required to be satisfied “as much as possible” for the planning purpose (as opposed to the exact satisfaction during on-line operations). These constraints are thus modeled as fuzzy relations following [5] and [6].

System Demand Constraints. Total generation plus expected future purchases minus sales should be “essentially” greater than or equal to the demand at each hour:

$$\text{p(t)} \equiv \sum_{i=1}^{N} p_i(t) + \sum_{m=1}^{M} E\left[p_m(t)\right] - \sum_{n=1}^{N} E\left[p_n(t)\right] \tilde{\geq} P_d(t)$$

$$, t = 1, 2, .., T. \qquad (11)$$

The membership of the above fuzzy relation is assumed for simplicity to be piecewise linear as described by:

$$\mu_p(t) = \begin{cases} 1, & \mathrm{p(t)} \ge \mathrm{P_d}(t), \\ 1 - \dfrac{P_d(t) - p(t)}{\delta_p(t)}, & \mathrm{P_d(t) > p(t) \ge P_d(t)} - \delta_p(t), \\ 0, & \textit{otherwise}, \end{cases}$$

$$t = 1, 2, .., T. \qquad (12)$$

In the above, $P_d(t)$ is the "nominal" system demand, and $\delta_p(t)$ the maximum range of demand variations. Equation (2.5) states that the demand becomes less satisfied as p(t) decreases below $P_d(t)$ as indicated by the reduced membership.

System Reserve Constraints. The total reserve contribution of all the units should be "essentially" greater than or equal to the reserve required at each hour:

$$\mathrm{r(t)} \equiv \sum_{i=1}^{I} r_i(t) \mathrel{\tilde{\ge}} P_r(t),$$

$$t = 1, 2, .., T. \qquad (13)$$

The membership of the fuzzy relation is assumed to be:

$$\mu_r(t) = \begin{cases} 1, & r(t) \ge \mathrm{P_r}(t), \\ 1 - \dfrac{P_r(t) - r(t)}{\delta_r(t)}, & \mathrm{P_r(t)} > r\mathrm{(t)} \ge \mathrm{P_r(t)} - \delta_r(t), \\ 0, & \textit{otherwise}, \end{cases}$$

$$t = 1, 2, .., T. \qquad (14)$$

In the above, $P_r(t)$ is the "nominal" reserve requirement, and $P_r(t)$ - $\delta_r(t)$ the minimum acceptable reserve. Equation (14) states that the reserve becomes less satisfied as r(t) decreases below $P_r(t)$ as indicated by the reduced membership.

Beyond systemwide demand and reserve requirements, there are individual unit and transaction constraints as detailed in [7], [1], and [2].

D.3 Solution Methodology

To solve the above problem with manageable complexity, the problem is first transformed into a crisp one by using the symmetric approach for fuzzy optimization ([8]). The transformed problem is decomposed into individual unit and transaction subproblems by using Lagrangian Relaxation. Individual subproblems are then solved by using dynamic programming.

Converting Fuzzy Optimization to a Crisp One. Based on the symmetric approach for fuzzy optimization, the objective J in (11) should be "essentially smaller than or equal" to some "aspiration level" J_0:

$$J \mathrel{\tilde{\le}} J_0 . \qquad (15)$$

The membership of the above "fuzzy objective constraints" is assumed to be:

$$\mu_J = \begin{cases} 1, & \mathrm{J} \le J_0, \\ 1 - \dfrac{J - J_0}{\delta_J}, & \mathrm{J_0 < J} \le J_0 + \delta_J, \\ 0, & \textit{otherwise}. \end{cases} \qquad (16)$$

The aspiration level J_0 represents the desired total cost. A schedule becomes less acceptable as the cost increases above J_0 as indicated by the reduced membership, and the highest acceptable cost is $J_0 + \delta_J$. One candidate for J_0 is the cost of the crisp scheduling problem with lowest acceptable demand and reserve, and expected parameters for future transactions.

In the symmetric approach, the fuzzy demand, reserve, and objective constraints are desired to be satisfied simultaneously. The problem is thus to maximize the minimum degree of satisfaction z among all fuzzy constraints by properly scheduling thermal units and making good transaction decisions:

$$\max z, \text{ with } z \equiv \min \{\mu_J , \mu_p(t) , \mu_r(t)\},$$

$$t = 1, 2, .., T, \qquad (17)$$

subject to individual unit and transaction constraints. To make the problem approximately equivalent to the crisp case when all fuzzy constraints can be satisfied with membership 1, the problem is modified following [6] to:

$$\min b(z-2)^2 + J, \qquad \text{with } b >> J,$$

subject to (18)

$$z \le \mu_J, \qquad (19)$$

$$z \le \mu_p(t), \ t=1, 2, \ldots, T \qquad (20)$$

$$z \le \mu_r(t), \ t=1, 2, \ldots, T, \qquad (21)$$

$$0 \le z \le 1, \qquad (22)$$

and all individual unit/transaction constraints. Since (17) - (20) are additive in terms of individual unit and transaction variables, the problem is "separable." Lagrangian relaxation can thus be effectively applied.

Lagrangian Relaxation. Lagrangian relaxation is applied to relax "systemwide constraints" (18) - (20) with the Lagrangian obtained as:

$$L \equiv b(z-2)^2 + J + \lambda_J[J - J_0 - (1 - z)\delta_J]$$
$$+ \sum_{t=1}^{T} \lambda(t)\Big[P_d(t) - p(t) - (1-z)\delta_p(t)\Big]$$
$$+ \sum_{t=1}^{T} \mu(t)\Big[P_r(t) - r(t) - (1-z)\delta_r(t)\Big]. \qquad (23)$$

For a given set of multipliers, the following subproblems are obtained after re-grouping relevant terms:

Thermal subproblem for unit i:

min L_i, with

$$L_i \equiv \sum_{t=1}^{T} \left\{ (1+\lambda_J)\left[C(p_i(t)) + S_i(t) \right] - \lambda(t) p_i(t) - \mu(t) r_i(t) \right\}; \qquad (24)$$

subject to individual thermal unit constraints. This subproblem is very similar to its counterpart when no uncertainty is involved ([7]), with fuzziness modifying the cost coefficients. The subproblem can thus be solved by using dynamic programming presented in [7] with straightforward modifications.

Future purchase transaction m:

min L_m , with

$$L_m \equiv E\left[\sum_{t=1}^{T} (1+\lambda_J) c_m(t) p_m(t) - \lambda(t) p_m(t) \right], \qquad (25)$$

subject to $p_m(t) \le \bar{p}_m(t)$ and individual purchase transaction constraints. Parameter $\bar{p}_m(t)$ is the maximum purchase level offered at time t, and is a random variable.

Future sale transaction n:

min L_n , with

$$L_n \equiv E\left[\sum_{t=1}^{T} (1+\lambda_J) c_n(t) p_n(t) + \lambda(t) p_n(t) \right], \qquad (26)$$

subject to $p_n(t) \le \bar{p}_n(t)$ and individual sale transaction constraints. Parameter $\bar{p}_n(t)$ is the maximum sale level at time t, and is a random variable.

The treatment of a future purchase or sale transaction subproblem for the deterministic case can be found in [1] and [2]. The resolution of the fuzzy-stochastic case follows a similar procedure. By discretizing the probability distributions of transaction price and maximum level for a load period, a set of price-MW combinations can be obtained, each associated with a probability. Backward dynamic programming is performed for each possible price-MW combination within a load period. The cost for the load period is then obtained as the expected value of the costs at the first hour of the load period, considering all possible price-MW combinations of that period. These periods are then linked together by using stochastic dynamic programming to satisfy allowable transaction patterns. The two steps are performed iteratively backwards in time, starting with the last load period. Since the effects of uncertainties are summarized in the expected “optimal-costs-to-go” at the first hour of each load period, the computational requirements increase only linearly as the number of possible price-MW combination increases.

Fuzzy Membership subproblem:

min L_z , with

$$L_z \equiv b(z-2)^2 + \lambda_J \delta_J z + \sum_{t=1}^{T} \left[\lambda(t) \delta_{p(t)} + \mu(t) \delta_{r(t)} \right] \cdot z \qquad (13)$$

This subproblem can be solved analytically by minimizing a quadratic function.

The multipliers are updated at the high level by using the “reduced complexity bundle method” [9].

D.4 Numerical Testing Results

The algorithm was implemented in FORTRAN and C++ on a SUN Ultra Sparc 170 workstation. Testing is based on two data sets from Northeast Utilities Service Company (NU): February week 2, 1996 and February week 3, 1996. Each data set has a time horizon of one week (168 hours), and includes about 70 thermal units, 7 hydro units, and 1 large pumped-storage unit. Hourly demand is assumed to subject to 4% variations, and hourly reserve 7% variations. The solutions satisfy all the rules of New England Power Pool.

For comparison purpose, a deterministic version of the algorithm is also tested using the same simulation shell. This deterministic version requires demand and reserve to be satisfied crisply, and ignores future transaction opportunities. Simulations were then performed for both the new and the deterministic algorithms, 200 Monte Carlo runs each. In the simulation, the algorithm is run on a daily basis in a moving window fashion, each time simulating a period of one week to ten days. Uncertainties are realized on a daily basis, and realized demand and reserves are required to be satisfied crisply. Decisions regarding realized transaction opportunities are incorporated into the system load. Testing results are summarized in Tables 1 and 2, with optimal membership 0.786 for Data Set 1 and 0.805 for Data Set 2.

The following can be observed from Tables 1 and 2:

1. The new algorithm results in significant savings for both schedule costs (2.3) and simulation costs as compared to the deterministic one, indicating that a good balance is achieved between minimizing costs and hedging against uncertainties.
2. The difference between a schedule cost and the associated simulation cost for the new algorithm is consistently lower than that of the deterministic version, indicating that the new method is less susceptible to disruptions and it thus more robust.
3. The computation time of the new algorithm is only slightly higher (about 20-30%) than that of the deterministic version. Uncertainties are thus handled in a computationally manageable manner.

ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation under Grant ECS-9420972, Northeast Utilities Service Company, and UConn's School of Engineering. The authors would like to thank Mr. M. B. Gabris, Mr. G. Merchel, Mr. J. A. Palmberg, and Mr. C. Larson of Northeast Utilities and Mr. D. Zhang and Mr. K. Kasiviswanathan of UConn for their valuable suggestions and support.

Table 1. Simulation Results of February Week 2, 1996

	Fuzz-Stoch	Crisp	Difference
Future purchase trans.	10	0	10
Future sale trans.	7	0	7
High level iterations	59	48	11
Schedule cost ($)	8,412,205	8,451,957	-39,752*
CPU time (sec)	29	24	5
Ave. simu. cost ($)	8,422,835	8,487,126	- 54,289*
Difference ($) (sche - simu)	-10,630	-35,169	--

* Negative number indicates savings.

Table 2. Simulation Results of February Week 3, 1996

	Fuzz-Stoch	Crisp	Difference
Future purchase trans.	9	0	9
Future sale trans.	6	0	6
High level iterations	65	51	14
Schedule cost ($)	8,352,189	8,422,360	-70,171*
CPU time (sec)	35	28	7
Ave. Simu. cost ($)	8,335,353	8,381,127	- 45,774*
Difference ($) schedule - simulation	16,836	41,233	--

* Negative number indicates savings.

D.5 References

[1] L. Zhang, P. B. Luh, X. Guan and G. Merchel, "Optimization-Based Inter-Utility Power Purchase," *IEEE Trans. on Power Systems,* Vol. 9, No. 2, May 1994, pp. 891-897.

[2] B. Prasannan, P. B. Luh, H. Yan, J. A. Palmberg, and L. Zhang, "Optimization-Based Sale Transactions and Hydrothermal Scheduling," *IEEE Trans. on Power Systems,* Vol. 11, No. 2, May 1996, pp. 654-660.

[3] H. Yan, P. B. Luh, X. Guan and P. M. Rogan, "Scheduling of Hydrothermal Power systems," *IEEE Trans. on Power Systems,* Vol. 8, No. 3, Aug. 1993, pp. 1358-1365.

[4] X. Guan, P. B. Luh, H. Yan, and P. M. Rogan, "Optimization-based Scheduling of Hydrothermal Power Systems with Pumped-Storage Units," *IEEE Trans. on Power Systems,* Vol. 9, No. 2, May 1994, pp. 1023-1031.

[5] H. Yan and P. B. Luh, "A Fuzzy Optimization-Based Method for Integrated Power System Scheduling and Inter-Utility Power Transaction with Uncertainties," *to appear in IEEE Trans. on Power Systems*.

[6] X. Guan, P. B. Luh and B. Prasannan, "Power System Scheduling with Fuzzy Reserve Requirements," *IEEE Trans. on Power Systems,* Vol. 11, No. 2, May 1996, pp. 864-869.

[7] X. Guan, P. B. Luh, H. Yan and J. A. Amalfi, "An Optimization-Based Method for Unit Commitment," *International Journal of Electrical Power & Energy Systems,* Vol. 14, No. 1, pp. 9-17.

[8] R. Bellman and L. A. Zadeh, "Decision-Making in A Fuzzy Environment," *Management Science,* Vol. 17, 1970, pp. 141-164.

[9] R. N. Tomastik, P. B. Luh, "A Reduced-Complexity Bundle Method for Maximizing Concave Nonsmooth Functions," *Proceedings of the 35th IEEE Conference on Decision and Control*, Kobe, Japan, Dec. 1996, pp. 2114-2119.

Chapter 7 Optimization Techniques II

A. Generation Expansion Problem by Means of Multi-Objective Fuzzy Optimization

(by Hiroshi Sasaki, Junji Kubokawa at Hiroshima University)

A.1 Introduction

Electricity is the indispensable form of energy in modern societies; its demand has been increasing year by year. Furthermore, a widespread use of various advanced electronic apparatus intensifies the sheer need of supplying high quality electric energy. Generation facilities of a power system must be expanded adequately so that it will be able to meet future demand increase. Hence, generation expansion problem has occupied an important position in power system engineering field.

The generation expansion planning (GEP) has been so far defined as a problem to determine the amount of new generation facility to be constructed so that the sum of fixed and variable costs of generation facilities is minimized over a certain period of time. Conventionally, GEP has been formulated as a non-integer programming problem in which a continuous variable is allocated to each type of generating units [1-3]. One possible approach is to apply linear programming after linearizing the original problem [4]. Although the linearization might seem inaccurate, it is still valid as a first approximation to the problem covering a very long time range in which many uncertain factors should be taken into account. More authentic, there is an approach based on nonlinear programming [5].

However, as is already mentioned, the requirement for more reliable electric energy supply is becoming more and more strong as the society has become more information oriented. As much as the same as high system security, it is indispensable to take into consideration environmental impact caused by power generation. Therefore, GEP may be formulated more appropriately as a multi-objective optimization problem in which economy, system security, and environmental stress should simultaneously be taken into account. This is especially the case in the recent trend of system planning methodologies, typically known as IRP (Integrated Resource Planning). In addition, power exchange may be considered one of very effective means to future supply of electric energy.

As another specific feature of GEP lies in the fact that it is a problem covering a long time span, well exceeding a decade or two. This means that GEP must postulate many assumptions which will be changing drastically or possess much uncertainties during the planning period. In reality, planning engineers must make up many alternative plans to allow for these uncertainties and future fluctuations of basic parameters such fuel cost, demand forecast. The decision maker must select one particular plan out of the thus provided alternatives based on his/her subjective judgment on many ambiguous factors. Thus, the incorporation of uncertainties has been again the recent trend in GEP [7] and an approach that can effectively handle such uncertainty is definitely necessary for solving GEP.

In this section, we shall take into consideration three objectives in GEP, that is, economy, supply reliability, and environmental impact by assuming all the variables can take continuous values. Conventionally, as to supply reliability, the reserve rate of about 6 - 10 [%] of its peak demand is assumed so as to accommodate inherently unforeseen faults or sudden loss of generation. However, in GEP that handles a long range planning, fluctuations on load demand forecast must be securely integrated. Therefore, two different concepts on supply reliability are needed: one is a conventional reserve rate as mentioned above and the other is supply reserve margin which stems from fluctuations in demand forecast to higher side at the target year. The latter is taken as an index to express supply reliability.

In order to minimize environmental impact from thermal generation units, it is natural to restrain the amount of gas emission which has been treated as constraints in conventional GEP. However, this critical factor must be treated equally with economy as well as supply reliability. In this sense, we shall introduce an environmental index to make compromise with the other objectives.

Multi-objective optimization problems generally cause much difficulty in the sense that which objective is more important than others and, if so, to what extent. In conventional mathematical methods for this class of problems, solutions of a multi-objective optimization problem are given as Parete solutions which consist of uncountable solution points. The decision maker must anyhow decide or select one specific solution out of the countless solutions by considering various factors relating to the planning. In general, the decision will be made based on his/her preference, experience, or linguistic judgment. Therefore, it may be said that the decision process is done on the basis of rather vague judgment.

Fuzzy mathematical programming has been developed significantly in recent years so that it can solve a class of multi-objective optimization problems with ambiguous or fuzzy constraints as well as objectives. Above all, fuzzy linear programming is a very effective method of making coordination among many conflicting or trade-off objectives. The coordination will be done through the shape of membership functions assigned to objectives and also to constraints. If the goal of a certain objective is not thought of much, this must be adjusted by redefining the associated membership function. Furthermore, it is advantageous to treat future demand prediction as fuzzy number. As already mentioned, the role of GEP under much uncertainty is to provide the decision maker with a set of

A.2 Fuzzy Linear Programming

In real world planning problems, their objectives and constraints are seldom rigid or crisp but rather vague in the degree of attainment. Fuzzy linear programming is a suitable mathematical tool to deal with such optimization problems with ambiguous objectives and constraints. The conventional multi-objective linear programming problem with k objectives may be formulated as follows:

$$\left.\begin{array}{ll}\text{minimize} & \mathbf{z}(\mathbf{x}) = \mathbf{C}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x} \le \mathbf{b} \\ & \mathbf{x} \ge \mathbf{0}\end{array}\right\} \quad (1)$$

where $\mathbf{z}^T(\mathbf{x}) = \left[z_1(\mathbf{x}),\ z_2(\mathbf{x}),\cdots,\ z_k(\mathbf{x})\right]$, $\mathbf{C} = \left[\mathbf{c}_1,\ \mathbf{c}_2,\cdots,\ \mathbf{c}_k,\right]$, $A^T = \left[\mathbf{a}_1,\ \mathbf{a}_2,\cdots,\ \mathbf{a}_m\right]$, and $\mathbf{x},\ \mathbf{c}_i,\ \mathbf{a}_i \in \Re^n$.

In general, the above problem has an infinite number of optimal solution points known as Parate optimal solutions. That is, if it is desired to improve a certain objective, this makes some other objectives deteriorate.

In real world problems, the constraints are mostly given as fuzzy quantities, in other words, by ambiguous or soft constraints. It is also true that the objectives are rarely need to be minimized or maximized in the strict sense, but they should attain values less than some taget ranges if to be minimized. Thus, linear programming with fuzzy objectives and constraints may be formulated as:

$$\left.\begin{array}{ll}\text{minimize} & \mathbf{z}(\mathbf{x}) = \mathbf{C}^T\mathbf{x} \lesssim \mathbf{z}^{\mathrm{o}} \\ \text{subject to} & A\mathbf{x} \lesssim \mathbf{b} \\ & \mathbf{x} \ge \mathbf{0}\end{array}\right\} \quad (2)$$

Here, symbol $\lesssim$ denotes fuzzy inequality and is used to express both fuzzy objectives and constraints. For instance, $A\mathbf{x} \lesssim \mathbf{b}$ means the left-hand term is roughly less than the right-hand term.

Equation (2) means that objectives and constraints are enforced by fuzzy inequality in fuzzy linear programming and hence can be treated equally. Fuzzy linear programming is based on the fuzzy decision principle proposed by R.E. Bellman and L.A. Zadeh. According to the principle, the fuzzy decision is defined as the intersection of fuzzy objectives (goals) and fuzzy constraints. Let us define the following notations for the membership functions of the k objectives and m constrinats as:

$$\left.\begin{array}{ll}\text{Objectives}: & \mu_{g_1}(x),\ \mu_{g_2}(x),\cdots,\mu_{g_k}(x) \\ \text{Constraints}: & \mu_{C_1}(x),\ \mu_{C_2}(x),\cdots,\mu_{C_m}(x)\end{array}\right\} \quad (3)$$

Then, the membership function of the fuzzy decision can be defined as

$$\mu_D(x) = \min(\mu_{g_1}(x), \mu_{g_2}(x),\cdots,\mu_{g_k}(x), \mu_{C_1}(x),\cdots,\mu_{C_m}(x)) \quad (4)$$

For convenience of further discussions, we shall introduce the following notations

$$\mu_g(x) = \mu_{g_1}(x), \mu_{g_2}(x),\cdots,\mu_{g_k}(x) \quad (5)$$

$$\mu_C(x) = \mu_{C_1}(x), \mu_{C_2}(x),\cdots,\mu_{C_m}(x) \quad (6)$$

By using these notations, the fuzzy decision can be defined as

$$\mu_D(x) = \min(\mu_g(x), \mu_C(x)) \quad (7)$$

The proposal of R.E. Bellman and L.A. Zadeh as a decision making in the fuzzy decision is to select x that maximizes membership function $\mu_D(x)$. This is expressed mathematically as

$$\mu_D(x^*) = \max_{x\in X} \mu_D(x) = \max_{x\in X}\left\{\min(\mu_g(x), \mu_C(x))\right\} \quad (8)$$

where X denotes the set of alternatives which consist of means or actions that can be taken in fuzzy decision.

According to fuzzy mathematical programming, (8) can be transformed to the following maximization problem:

$$\sup_{x\in X} \mu_D(x) = \sup_{\alpha\in[0,1]} \min[\alpha,\ \sup_{x\in C_\alpha} \mu_g(x)] \quad (9)$$

where C_α means the α-level set of the fuzzy constraints and expressed as

$$C_\alpha = \left\{x \middle| \mu_C(x) \ge \alpha\right\} \quad (10)$$

If $\varphi(x) = \sup_{x\in C_\alpha} \mu_g(x)$ is continuous with respect to α, the following holds:

$$\sup_{x\in X} \mu_D(x) = \sup_{x\in C_{\bar\alpha}} \mu_G(x) = \bar\alpha \quad (11)$$

Therefore, the fuzzy mathematical programming problem has reduced to tan ordinary methematical programming problem. That is, once $\bar\alpha$ is obtained, the remaining is to maximize $\mu_G(x)$ under a crisp constraint set $C_{\bar\alpha}$.

The algorithm shown in Fig. 1 is to obtain the maximum value of α for which the degree of satisfaction of each objective is greater than or equal to. Note that a suitable membership function is postulated for each objective depending on its relative importance. Since the minimum degree of satisfaction among the objectives is maximized by this algorithm, it is one of algorithms of realizing usual max-min operation of fuzzy objectives. In Fig. 1, first an optimization problem with single objective $z_i \ (i = 1,\ 2, \cdots,\ k)$ is solved separately to evaluate the ideal fluctuation range $a_{0i} < z_i < a_{1i}$. Based on this result, we define a membership function to express the degree of satisfaction for each objective. After postulating membership functions, set the initial value of α-level. Then, select an arbitrary objective function, say z_i, as the objective and the remaining objective functions are transformed to constraints as in the following:

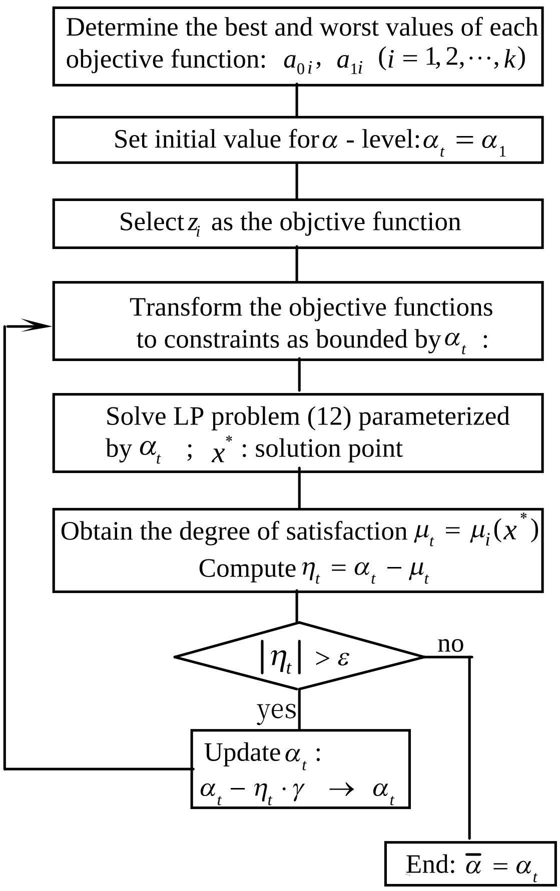


Fig. 1 Algorithm to solve fuzzy LP problems

$$\left.\begin{array}{ll} \text{minimize} & z_i = \mathbf{c}_i^T \mathbf{x} \\ \text{subject to} & A\mathbf{x} \le \mathbf{b} \ ; \ \ \mathbf{x} \ge \mathbf{0} \\ & z_j \le a_j \ \ \text{: minimization} \\ & z_j \ge a_j \ \ \text{: maximization} \\ & a_j = \mu_j^{-1}(\alpha_t) \ ; \ \ j \ne i \end{array}\right\} \tag{12}$$

That is, if a certain objective is to be minimized, it is constrained from downward by $a_j = \mu_j^{-1}(\alpha_t)$ that is determined by the degree of satisfaction α_t, and vice versa. Then, linear programming problem (2) is solved and the value of each membership function is respectively compared with α_t. If the discrepancy exceeds ε, α_t and therefore the upper or lower limit a_j is updated; the process continues until the error becomes negligible. The obtained α_t is the optimal. If the decision maker desires to further improve the degree of satisfaction, he must check and alter the specifications of some membership functions.

In fuzzy linear programming, the vagueness of the decision maker in selecting an adequate plan is reflected as the degree of satisfaction through the shapes of membership functions. Therefore, how to set up a membership function may characterizes constraints as well as objectives. Depending on the strictness of objective attainment, we shall assume the three kinds of membership functions as shown in Fig. 2. Here, type S is for the case of low priority, type H is for more stringent case, and type N is linear membership function. In this Figure, relationship between the value of an objective function a and the degree of satisfaction α is shown for the linear membership function.

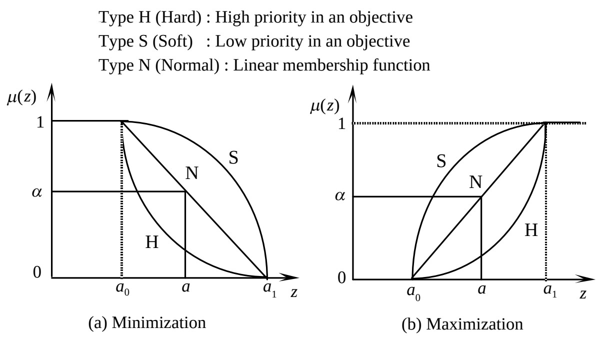


Fig. 2 Membership function

A.3 Formulation of Generation Expansion Planning

GEP can be defined as a problem of determining the kind and capacity of generation technologies to be installed and transmission networks to support power exchange between areas. We have postulated the following assumptions in our study:

(1) Fuel cost is the same among areas under consideration, that is, the cost is the same for the same kind of generation technology.
(2) Each unit has no time delay in its start-up and shutdown (complete follow-up capability for demand fluctuations)
(3) Each unit can be operated without any fault
(4) Merit order (priority order in start-up) is predetermined
(5) The load duration curve, maximum demand and spinning reserve are given.
(6) The load duration curve which is the sum of those of all areas is assumed to consist of five levels as shown in Fig. 3.

Here, the last assumption is necessary to make the problem be solved by linear programming.

The constraints and objective functions are explained in the below. Subscripts m, i, k attached to symbols denote the load level, area, and the kind of generation technology, respectively. The period of the planning is 10 years and four kinds of generations are assumed.

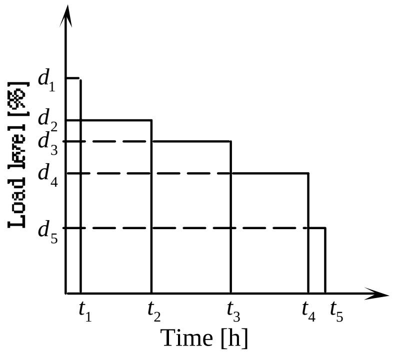


Fig. 3 Approximated load during curve

[Constraints]

(1) Constraints on generation capacity

This is concerned with the maximum generation capacity of the total system relative to the maximum load demand and may be expressed as

$$\sum_i \sum_k (x_{ik} + \Delta x_{ik}) \ge (1+r) \sum_i (D_i + \Delta D_i) \tag{13}$$

where

x_{ik} : Capacity of the existing technology k in area i

Δx_{ik} : Capacity of newly installed technology k in area i

D_i : Minimum forecasted load demand in area i

ΔD_i : Change in Forecasted load demand in area i

r : Reserve rate

(2) Constraints on the output share of each generation technology

The sum of the output of each generation technology must be equal to the sum of power exchange and load demand at each load level.

$$\sum_k Y_{mik} = \sum_{i,j} (L_{mij} + \Delta L_{mij}) + d_{mi} + \Delta d_{mi} \;\; ; \;\; L_{mij} = -L_{mji} \tag{14}$$

where

Y_{mik} : Generation share of technology k at load level m in area i

L_{mij} : Power exchange between areas i and j through the existing transmission line at load level m

ΔL_{mij} : Power exchange between areas i and j through the new transmission line at load level m

d_{mi} : Forecasted load demand at area i at load level m

Δd_{mi} : Change in forecasted load demand at load level m in area i

(3) Constraints on the output of each generation technology

At each load level, the output of each generation cannot exceed its installation capacity.

$$\sum_i Y_{mik} \le \sum_i (x_{ik} + \Delta x_{ik}) \tag{15}$$

(4) constraints on power exchange

The amount of power exchange between two areas must not exceed the transmission capacity of the transmission line.

$$|L_{mij}| \le L_{ij} \;\; ; \;\; |\Delta L_{mij}| \le \Delta L_{ij} \tag{16}$$

where

L_{ij} : Transmission capacity of the existing line between areas i and j

ΔL_{ij} : Transmission capacity of the new line between areas i and j

(5) Constraints on the capacity of new installation

There are certain constraints on the capacity of newly installed generation technology and transmission lines. This is especially the case for nuclear plants. Since any nuclear unit is operated at a constant output at present and hence it cannot follow up load variations. In other words, the maximum total capacity of nuclear units should be prescribed so that they could be used as base loading.

$$\underline{\Delta X_{ik}} \le \Delta X_{ik} \le \overline{\Delta X_{ik}} \;\; ; \;\; \Delta L_{ij} \le \overline{\Delta L_{ij}} \tag{17}$$

where

$\underline{\Delta X_{ik}}$, $\overline{\Delta X_{ik}}$: Lower and upper limit of new installation of technology k in area i

$\overline{\Delta L_{ij}}$: Upper limit of the capacity of new transmission line between areas i and j

(6) The lower and upper limits of fluctuations in forecasted load

It is convenient and natural to anticipate that fluctuations between the actual load demand and the forecasted one at the planning period may lie in a range.

$$\underline{\Delta D_i} \le \Delta D_i \le \overline{\Delta D_i} \tag{18}$$

where $\underline{\Delta D_i}$, $\overline{\Delta D_i}$: Lower and upper limit of fluctuations in demand forecast

[Objective Functions]

(1) Economy index

As an economy index, we shall consider the sum of the annual investment costs of generation plants and transmission lines, fuel cost and purchase cost through interchange. It is assumed that investment costs are in proportion to its capacity and fuel cost to the amount electric energy produced; the cost of power purchase is assumed to be proportional to the capacity of inter-tie transmission line. Note that the investment cost is converted to the present value. Then, the economy index is expressed as:

$$F_1 = \sum_i \sum_k f_{ik} \Delta x_{ik} R + \sum_m \sum_i \sum_k v_k Y_{mik} T_m R + \sum_i \sum_j c_{ij} \Delta L_{ij} R + \sum_m \sum_i \sum_j b_{ij} (\Delta L_{mij} + L_{mij}) T_m R \quad (19)$$

where

f_{ik} : Per unit annual investment cost of generation technology k in area i

R : Conversion coefficient to the present worth

v_k : Fuel cost of generation technology k

T_m : Duration of load at level m

c_{ij} : Per unit annual unit investment cost of transmission lines between areas i and j

b_{ij} : Per unit annual cost of power exchange between areas i and j

(2) Environment index

Although thermal plants are emitting NO_x and SO_x as well as CO_2, we shall take into consideration only CO_2 for simplicity of treatment. The amount of CO_2 emission is assumed to be in proportional to generated energy (MWh) and expressed by

$$F_2 = \sum_m \sum_i \sum_k e_k Y_{mik} T_m R \quad (20)$$

where e_k denotes emission coefficient of CO_2 of technology k

(3) Supply reserve margin index (to be maximized)

This index is reflecting fluctuations in load forecast and to be maximized to keep a high supply reliability. As this certainly brings about much higher cost, there is a certain limit which will be determined by fuzzy coordination. We shall adopt the sum of fluctuations in load demand in all areas as the index:

$$F_3 = \sum_i \Delta D_i \quad (21)$$

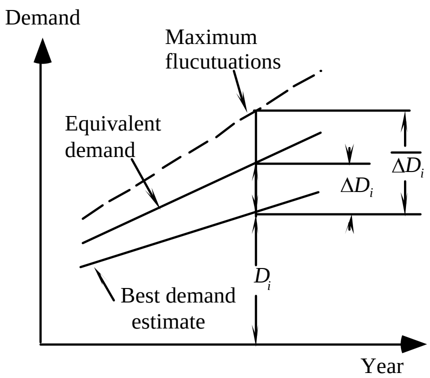


Fig. 4 Fluctuations in load forecast

This index may be explained by utilizing Fig. 4, in which three lines are depicted. The line at the bottom means the best (and possibly minimum) demand forecast at the planning year; the line at the middle shows fluctuations against the best demand forecast that actually occur in reality. The fluctuation denoted by ΔD_i in the Figure can be regarded as supply reserve margin from the standpoint of supply reliability. However, it is assumed that possible fluctuations are bounded by the line at the top, that is, by $\overline{\Delta D_i}$. Here, we shall designate $D_i + \Delta D_i$ as the equivalent demand.

A.4 Discussions on Numerical Simulations

(1) Test Systems

Fig. 5 shows a test system which is used in numerical simulations to evaluate the validity of the proposed fuzzy approach. Generating plants and transmission lines depicted in doted lines signify future possible installations and those in solid lines existing plants and lines. Four kinds of generation sources are considered, that is, nuclear (H), coal (C), oil (O) and LNG (L). Load demands at the reference year and demand forecasts at the target year in each area are given in Table 1.

Table 1 Load demand in each area

Area	Load demand in the reference year [MW]	Forecasted load in the target year [MW]
1	1300	1800
2	2500	3800
3	3600	5200
4	2500	3100

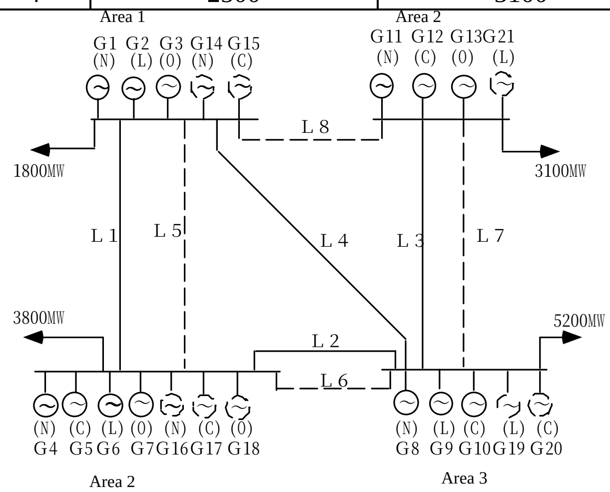


Fig. 5 Test System

(2) Coordination Between Economy and Environment Indices

In this subsection, we shall discuss the effect of introducing the environmental index in GEP with interconnections. In the proposed method, to what extent the environmental impact is taken into account in GEP depends on how to set up the associated membership function.

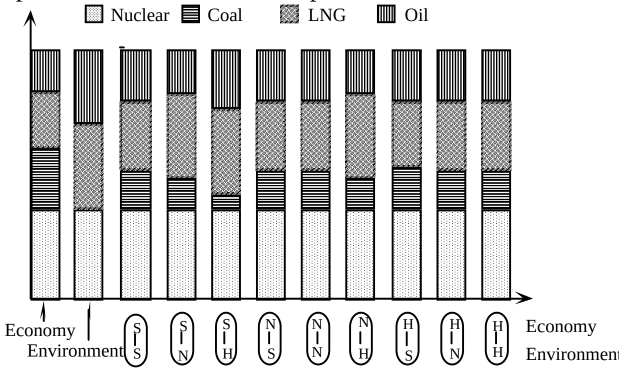


Fig. 6 Best mix for the case of the fuzzy coordination between the economy and environment index

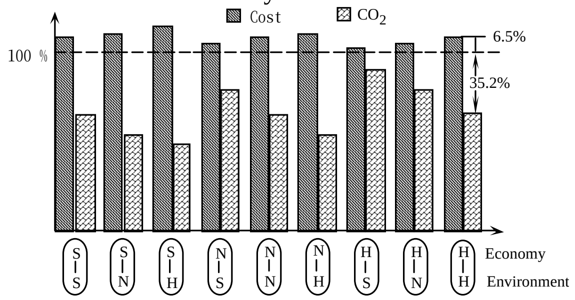


Fig. 7 The cost and CO_2 emmison for the case of fuzzy coordination between the economy and environment index

In the first place, the generation mix with only the economy index considered is shown at the left column in Fig. 6. In this case, nuclear and coal-fired thermal units are mostly constructed because of their economical advantage over other types of generation. Furthermore, power interchange via the exiting lines is not incorporated so much in this case. In the next, we shall consider another extreme case where only environmental index is taken into consideration, though this is quite impractical and just for the purpose of comparison. It is clear from the second to the left column in Fig. 6 that coal units which emit a large amount of CO_2 are not constructed at all. This is replaced by increases in oil and LNG units, while the share of nuclear units is practically unchanged due to the constraint. Although a large amount of power exchange through the exiting lines is incorporated in the planning, it generally depends on the initial allocation of generation technologies to each area. The cost and the amount of CO_2 emission in this case are respectively 120 [%] and 46.4 [%] of those in the case of economy index only.

Simulation results for the case of coordination between the economy and environmental indices are summarized in Figs. 6 and 7; the former depicts the generation mix (share of each generation technology) at the target year and the latter shows the variations of the cost and CO_2 emission taking the case with only the economy index considered as the reference. It should be noted that the share of nuclear plants remains the same in Fig. 7 irrespective of difference in membership function settings as a result of constraint 5. This is because nuclear plants cannot change their outputs. In general, coal-fired units are not friendly to the environment and hence as the requirement for reducing environmental impact is strengthened, the rate of coal plants reduces (every three cases from the left). Of course, the degree of reduction in coal units is largely affected by the membership function corresponding to the economy index. Fig. 8 fortify these observations. In fact, for the case where both economy and environment have H-type membership functions, the cost increases by 6.5 [%], but the emission decreases by 35.2 [%] compared with the case with only economy index taken into account.

(3) Coordination between Economy and Supply Reserve Margin Index

In this subsection, we shall discuss results of the coordination between economy and supply reserve margin. Fig. 8 shows how the cost and the supply research margin index change, in which the both indices are expressed as the ratio to that of the case with only the economy index considered. In Fig. 9, the capacity of newly constructed generating plants and the amount of power interchange are shown for different combination of membership functions. It is obvious that when the economy is weighed more (type H), the cost and inevitably the equivalent demand reduces. Also, we can observe that as the capacity of new generating plants decreases, power interchange increases significantly (see every 3 columns in Fig. 9). These Figures demonstrate that economy and supply reserve margin are in a trade-off relationship. Therefore, it may be concluded that by flexibly changing the shapes of the membership functions, the proposed system can provide a set of alternatives that reflects the intention of the decision maker.

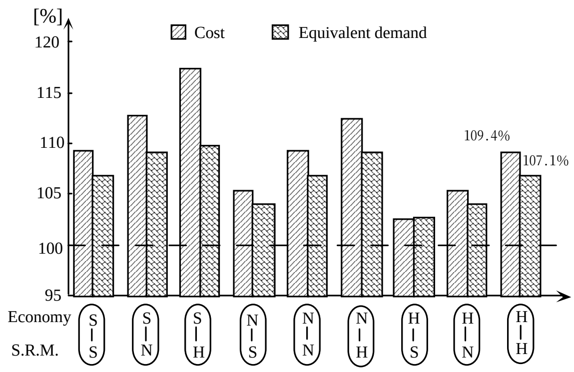


Fig. 8 The cost and equivalent demand for the case of fuzzy coordination between the economy and supply reserve margin index (S.R.M)

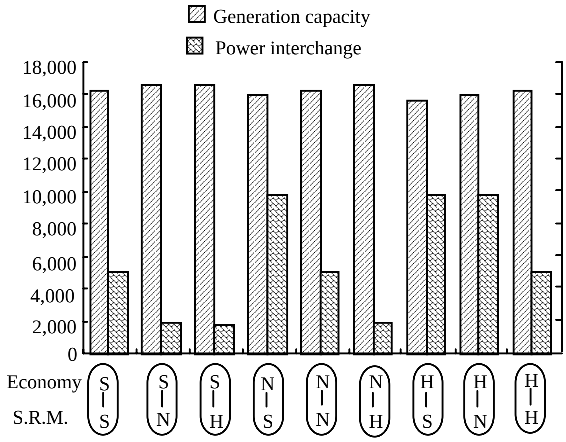


Fig. 9 The generation capacity and power interchange for the case of fuzzy coordination between the economy and supply reserve margin index (S.R.M.)

(4) Effect of Fluctuations of Demand Forecast

As is mentioned in the introduction, there is not a small probability that the predicted demand fluctuates significantly. If it fluctuates downward during the planning period, no reliability problem does occur though the expansion planning is quite uneconomical. A serious problem does occur if the estimate deviates upward. Since it is very likely to occur, GEP must be able to handle fluctuations in demand forecast. As an effective means of handling this situation, the predicted load demand is fuzzified, that is, it is regarded as a fuzzy number.

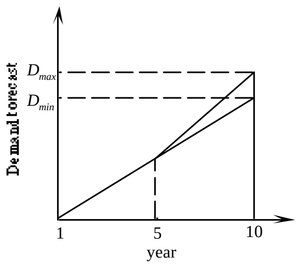


Fig. 10 Increase in load demand after the fifth year

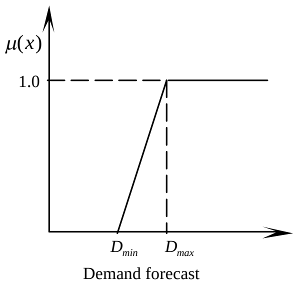


Fig. 11 Membership function

We assume that the load demand increases unexpectedly after the fifth year as shown in Fig. 10. Here, D_{min} is the same as given in Table 1 and D_{max} is the actual demand in the target year. The rate of the fluctuations, defined by $(D_{max} - D_{min})/D_{min}$, are postulated to be 3, 5, 7, and 10 [%]. Fig. 11 shows the membership function corresponding to load increases against the original forecast. For unexpected load growth after the fifth year, it is assumed that coal and gas turbine units could be constructed due to their relatively short lead times.

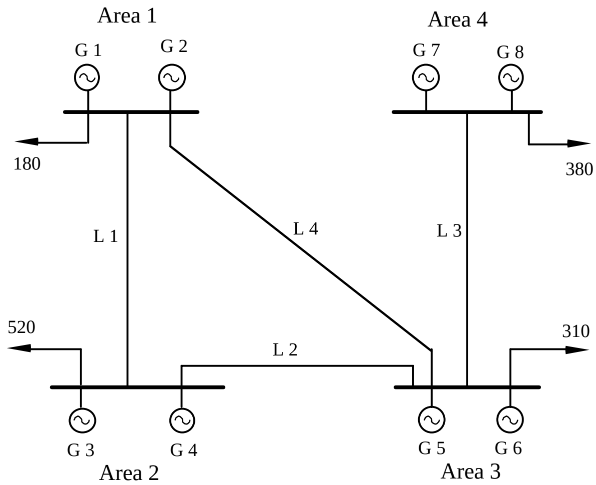


Fig. 12 Possible generation plants to be constructed for demand increase

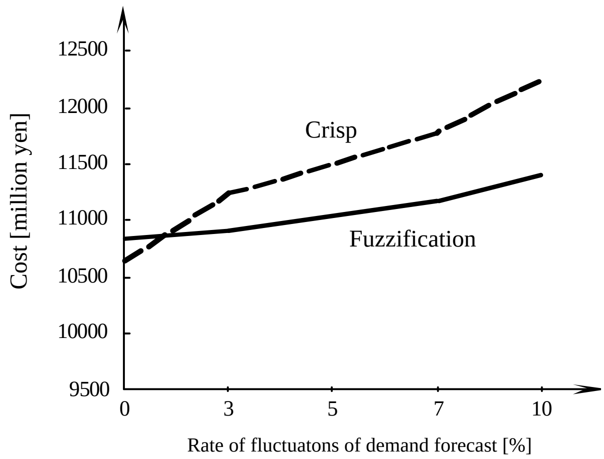


Fig. 13 Relationship between the cost and the rate of fluctuations in demand forecast

By making use of the above data, simulations have been carried out for cases with and without treating load forecast as fuzzy numbers. Fig. 13 shows the total cost versus the rate of fluctuations in the load forecast. In this Figure, costs corresponding to the crisp case are computed as the sum of the cost for the expected minimum, D_{min} and additional cost incurred by the unexpected load growth. On the other hand, in case of the fuzzification, the planning is completed at the outset by taking into consideration both D_{min} and D_{max} . If the demand forecast would not fluctuate, the cost in the crisp case is clearly the optimum. However, with an increase in fluctuations, the cost increases significantly compared with the fuzzification case. In general, since a rigorous forecast of load demand for a long time range which is one of the most important basic data for GEP, these results shown in Fig. 13 have verified a clear advantage of treating load forecast as fuzzy numbers.

A.5 Concluding Remarks

In this section, the generation expansion planning (GEP) has been formulated as a multi-objective optimization problem, in which economy, supply reliability and environmental impact are taken as objectives. Also, power interchange is included in the formulation in order to allow for a multi-area system. GEP thus formulated has been successfully solved by the fuzzy linear programming method. In the proposed method based on fuzzy coordination, it is possible to make up a set of alternative plans that take into consideration trade-off among the three objectives. Therefore, this can be a truly useful tool for the decision maker.

A.6 References

[1] F. Beglari and M.A. Laughton, "The combined costs method for optimal economic planning of an electrical power system," IEEE Trans. on Power Apparatus and Systems, vol. PAS-94, No. 6, pp.1935-1942, Nov./Dec. 1975

[2] W. Rutz, et al, "Sequential objective linear programming for generation planning," *ibid*, vol. PAS-98, no. 6, pp.2015-2021, Nov./Dec. 1979

[3] N. Levin and J. Zahavi, "Optimal mix algorithms with existing units," *ibid*, vol. PAS-103, No. 5, pp.954-962, May 1984

[4] J.K. Delson, et al, "Linear programming applications to power system economics planning and operations," IEEE Transactions on Power Systems, vol. 7, pp. 1155-1163, No. 3, August 1992

[5] Andres Ramos, et al, "A nonlinear programming to optimal static generation expansion planning," IEEE Transactions on Power Systems, vol. 4, pp. 1140-1146, No. 3, August 1989

[6] H. Sasaki, J. Kubokawa, M. Watanabe, R. Yokoyama and R. Tanabe, "A solution of generation expansion problem by neural networks," The First International Forum on Applications of Neural Networks to Power Systems, pp.219-224, July 23-26, 1991, Seattle, USA

[7] B.G. Gorenstin, et al, "Power system expansion planning under uncertainty," IEEE Transactions on Power Systems, vol. 8, pp. 129-136, No. 1, February 1993.

B. A Solution Method of Multi-objective Optimal Power Flow by Means of Fuzzy Coordination

(Junji Kubokawa, Hiroshi Sasaki, Hiroshima University; Ryuichi Yokoyama, Tokyo Metropolitan University)

B.1 Introduction

In recent years, more stringent requirements have been imposed on electric utilities. Personnel in charge of system operations is requested to determine optimal system states that satisfy versatile operational constraints. As a powerful means of solving this class of problems, extensive studies on the "Optimal Power Flow (OPF)" have been undertaken [1,2]. It should be noted here that OPF is to optimize just one specific objective, or single performance index. In general, to attain an optimal operation of a large scale system,typified by an electric power system, multiple objectives with different natures must be taken into consideration. For a power system, economic operations, supply reliability, security and minimal impact on environments are typical objectives to be satisfied. It is obvious that trade-off among these objectives is impossible because of their differences in nature. For instance, to improve the security certainly deteriorates system economy due to a large amount of investments. This fact is stated that the objectives are non-commensurable.

Unfortunately, conventional optimization techniques are not suitable to obtain the optimal solution which simultaneously optimize a variety of objectives. One conceivable approach using conventional methods is to convert a multi-objective problem into a single objective problem by assigning distinct weights to each objectives, thereby allowing for relative importance among goals [3]. However, this artifice is not totally satisfactory since different objectives cannot be evaluated under a common measure and there are no rational basis of determining adequate weights. The main purpose of the optimal power dispatch problems have so far been confined to minimize the total generation cost of a power system. However, in order to meet environmental regulations enforced in recent years, emission control has become one of important operational objectives. In this problem, the amount of NOx emission, which is in proportion to the active power output of a generator, is selected as an evaluation criterion, and the minimum emission is sought within a small region around an economically feasible operating point [4,5].

System security is another essential factor in power system operation and also in system planning. To be specific, it is very important to maintain good voltage profiles and to limit line flows within prescribed upper bounds. In security analysis, a series of anticipated contingencies are assumed to predict possible overloading or excessive voltage deviations. Then, a security index as a function of overloads and/or voltage excursions will be minimized by some preventive control actions.

When permissible limits of emission and overloads are clearly specified in a power system under study, these quantities could be incorporated into the OPF as operational constraints. However, in system planning studies, these limits posed on emission or overloads would be very ambiguous, thus making such treatment difficult. Also, in actual system operations, it is necessary to maintain the system at a proper security and emission level even when generator or transmission line tripping do occur. To attain this goal, system operating points should not be at constraint limits but needs some operational margin. Furthermore, operation indices mentioned herein are in conflicting trade-off relations, successful optimization cannot be attained through any of conventional optimization approaches.

In power flow optimization problems, there exist a number of objectives to be achieved which inherently have different characteristics, and hence conflicting relations hold among these objectives. To be specific, indices associated with economy, reliability and environment protection are non-commensurable in their nature. Moreover, there is no invariant priority order among these indices considering drastic changes in circumstances surrounding electric utilities. In these cases, some personnel (referred to as "decision maker: DM") must decide which is optimal based on his/her subjective judgments. Therefore, it is indispensable to grasp quantitatively these trade-off relations in order to obtain the optimal solution in an objective way.

Here, a brief review will be given on the following three alternative approaches to deal with optimization problems having multiple objectives that are in trade-off relations and non-commensurable:

(1) Scalarization method approach.
(2) Goal programming method [6].
(3) Fuzzy coordination approach.

The first approach may be classified to the following three different methods:

(a) An approach in which only a specific objective to be regarded as the most essential is taken as the performance index, while the others are treated as constraints.
(b) An approach in which a scalar composite performance index is made up by properly weighting each objectives.
(c) An approach in which the most essential objective is taken as the primal performance index and the remaining objectives are processed in each sub-problems as respective performance indices [7].

Approach (a) works well only when the goal to be attained is well defined and priority order among objectives is given a priori. This treatment, however, encounters a severe difficulty in case trade-off relations among contradictory objectives, such as between reliability and economy, are vaguely given. As to approach (b), it is quite easy to optimize a scalar performance index composed of as a linear combination of involved objectives. However, it is of serious doubt to assign weights in homogeneous manner to objectives being not commensurable. Furthermore, the meaning of weighting factors is difficult to justify. On the other hand, the main advantage of approach (c) is in that it can clearly handle trade-off relations among conflicting objectives. Nevertheless, there is no definite guide to select which one should be selected as the most desirable from a group of optimal solution candidates. This method, referred to as the ε-constrained method, is very effective to obtain a set of non-inferiority solutions (or, Pareto-optimal solutions). Here, the concept of the non-inferiority implies that when an arbitrarily chosen index is to be improved, some other indices deteriorate more than the improvement gained in the selected index. Thus, the originally selected performance index gives no clue to choose the optimal solution.

In the goal programming, DM must set up goals or aspiration levels for the objective functions and minimizes deviations from the goals. This is to pursue a satisfaction of objectives rather than optimization, and in a sense very similar to the fuzzy coordination approach.

The fuzzy coordination approach is to maximize the degree of satisfaction of DM on each objective. Since a multi-objective optimization problem has uncountable solutions, DM must decide one specific solution point by his own decision. This is accomplished by postulating properly a membership function to each objective according to its importance to DM. In this approach, the degree of satisfaction will be improved step by step by updating or changing the membership functions.

B.2 Multi-objective Optimal Power Flow

Multi-objective Optimization Problem

A multi- objective optimization problem is to minimize simultaneously p objective functions on X, a set of feasible solutions, and may be formulated as follows:

$$\left.\begin{array}{ll} \textit{minimize} & \boldsymbol{f}(\boldsymbol{x}) \\ \textit{subject to} & \boldsymbol{x} \in X = \{\boldsymbol{x} \mid \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}\} \end{array}\right\} \qquad (1)$$

where $\boldsymbol{x}$: decision variable vector
X : a feasible set of $\boldsymbol{x}$
$\boldsymbol{f}(\boldsymbol{x}) = \{f_1(\boldsymbol{x}),\ f_2(\boldsymbol{x}), \cdots,\ f_p(\boldsymbol{x})\}^T$: vector objective functions
$\boldsymbol{g}(\boldsymbol{x})$: inequality constraints

In general, a complete optimal solution that simultaneously minimizes all of the multiple objective function does not always exist when the objective functions conflict with each other. Thus, the concept of "Pareto optimal solution" or "non-inferior solution" is introduced, in which the improvement of a particular objective function must cause deterioration of other objective functions. From the definition, the non-inferior solution set consists of an infinite number of points. In optimal power flow problems, it is important to get a unique solution which reflects the preference of Decision Maker (DM).

Evaluation Indices

Economy, environmental and security index, which can be regarded the most important subject in power system operations, are chosen as the evaluation indices.

(a) Economy index

The fuel cost of a thermal unit is an essential criterion for economic feasibility and can be impressed by:

$$F_1(\boldsymbol{y}) = \sum_{i=1}^{n} \left(a_i + b_i \cdot P_{Gi} + c_i \cdot P_{Gi}^2 \right) \quad (2)$$

where P_{Gi} : Generator output of generator i

$a_i,\ b_i,\ c_i$: cost coefficients

(b) Environmental impact index

Nitrogen-Oxide (NOx) emission is taken as the index from the viewpoint of environment conservation. The amount of NOx emission is given as a function of generator output:

$$F_2(\boldsymbol{y}) = \sum_{i=1}^{n} \left(\alpha_i + \beta_i \cdot P_{Gi} + \gamma_i \cdot P_{Gi}^2 + \delta_i \exp(\varepsilon_i \cdot P_{Gi}) \right) \quad (3)$$

where $\alpha_i,\ \beta_i,\ \gamma_i,\ \delta_i,\ \varepsilon_i$: NOx emission coefficients.

(c) Line overload index

Overloading in a transmission line can lead to system collapse in an extreme case. Hence, we adopt, as the security index, a weighted sum of line flow deviations of all transmission lines. Thus, the security index is expressed as

$$F_3(\boldsymbol{y}) = \sum_{i=1, j=1}^{n} w_{ij} \cdot \xi\left(PL_{ij}(\boldsymbol{y}) - PL_{ij}^{*} \right) \quad (4)$$

where $\xi(\boldsymbol{x}) = \begin{cases} \boldsymbol{x} & \boldsymbol{x} > 0 \\ 0 & \boldsymbol{x} \le 0 \end{cases}$

w_{ij} : Weighting factor

$PL_{ij}(\boldsymbol{y})$: Line flow from node i to j

$PL_{ij}(\boldsymbol{y})^{*}$: Nominal transmission capacity

Equality Constraints

Since an OPF solution must satisfy power flow equations at each node, they are treated as equality constraints.

$$P(\boldsymbol{y}) - P^S = 0 \quad (5)$$

$$Q(\boldsymbol{y}) - Q^S = 0 \quad (6)$$

Inequality Constraints

Inequality constraints must be introduced to take into consideration various kinds of operational limits. In this study, voltage magnitude at each node, active and reactive generator output and line flows are used as inequality constraints.

B.3 Interactive Fuzzy Multi-objective Optimal Power Flow

Interactive Fuzzy Multi-objective Programming

Considering the vagueness of evaluation criteria of the multi-objective optimization problem, the decision maker (DM) seems to have fuzzy goals as "each objective function will be substantially less than some value". These fuzzy goals for DM can be quantified by specifying membership functions to the corresponding objective functions. Once DM having specified the membership functions, a fuzzy optimal solution can be obtained by solving maximization problem of the sum of the membership functions. If DM cannot satisfy the obtained solution, the DM is required to change the shapes of membership functions interactively. The linear membership function and add-operator for the fuzzy decision set were adopted in this study.

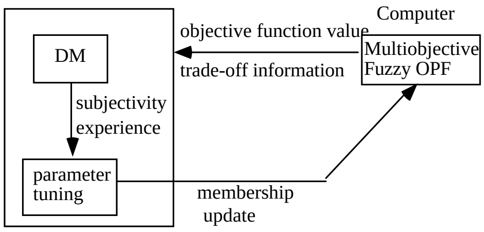
$$\begin{aligned} &\textit{maximize} \qquad DM(\boldsymbol{x}) = \sum_{k=1}^{p} \mu_k(\boldsymbol{x}) \\ &\textit{subject to} \qquad \boldsymbol{x} \in X = \{ \boldsymbol{x} \mid g(\boldsymbol{x}) \le 0 \} \end{aligned} \quad (7)$$

where $DM(\boldsymbol{x})$: fuzzy decision set

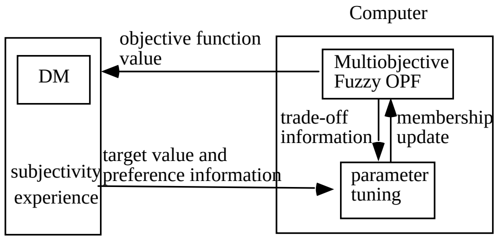
$\mu_k(\boldsymbol{x})$: the degree of satisfaction for objective function *k*

Although several methods have been proposed for fuzzy OPF[8,9], they assumed that some already tuned membership functions would be supplied by DM. It must be quite rare that such tuned membership functions are given a-priori in real applications. In order for DM to set up tuned membership functions more conveniently, we propose an interactive fuzzy OPF method, in which the membership function will be updated interactively by DM.

In the conventional interactive multi-objective optimization programming, the membership parameters have been updated by DM in accordance with the values of the objective functions and trade-off information (Figure 1(a)). However, because of complicated interactions between objective functions, it is impossible to predict how the solution will behave depending on changes in the membership functions. In the proposed method, only thing that DM has to do is to change the target values and preference information of the objective functions based on his experience; resulting membership updates will be carried out by the computer (Figure 1 (b)).



(a) Conventional Method



(b) Proposed Method

Fig. 1 The concept of the proposed interactive algorithm

Algorithm for Membership Function Update

(1) Change in the priority of objectives

In the proposed algorithm, the degree of satisfaction of DM is expressed as the distance from the target value (fuzzy goal) to the current Pareto optimal solution and the preference information on increase/decrease of the priority of each objective function.

Fig.2 (a) shows the linear membership function of this problem and Fig. 2 (b) explains how the priority of f_i can be changed by f_{i0}. If DM desires to improve the degree of satisfaction of objective f_i, the priority of f_i should be increased by either of the following two strategies.

(i) decrease parameter f_{i0}

(ii) increase parameter f_{j0} $(j \neq i, \ \ j = 1,\ldots,p)$

In this study, we have adopted strategy (ii) for updating membership functions. This is because the rate of trade-off among the objectives will be used to calculate the membership updates.

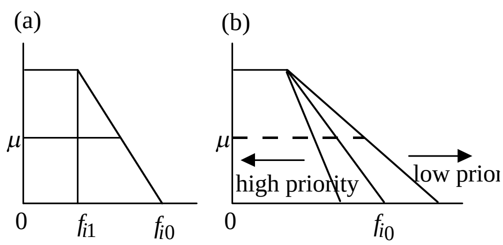


Fig. 2 (a) Linear membership function, (b) Tuning of me

f_{i0} : lowest acceptable value of the objective function

f_{i1} : satisfactory value of the objective function

(2) Update of the membership functions

In case of postulating linear membership functions, the trade-off rates among objective functions are in proportion to the trade-off rates of associated membership functions. This fact is the basic principle in the update calculation of the membership functions.

Now, let denote by α_{ij} a contribution of increasing/decreasing priority in accordance with DM's request:

$$\alpha_{ij} = \beta \cdot \frac{\partial f_i(\boldsymbol{x})}{\partial f_j(\boldsymbol{x})} \tag{8}$$

where $\beta > 0$: request for increasing priority of f_i

$\beta < 0$: request for decreasing priority of f_i

The total membership update for objective $f_j(x)$ is given by the sum of each contribution α_{ij} as

$$\alpha_j = \sum_{i=1}^{p} \alpha_{ij} \tag{9}$$

and the new membership parameter becomes

$$f_{j0} = f_{j0} + \alpha_j \tag{10}$$

(10).

B.4 Solution Algorithms

The followings are the solution algorithms for the fuzzy interactive multi-objective OPF.

Step 1: Calculate $f_i^{\min}$ and $f_i^{\max}$ of objective function $f_i(\boldsymbol{x})$; $i = 1, \cdots, p.$.

Step 2: Ask the DM to select the initial value of membership parameter f_{i0}, f_i. and target value μ_i^{target}.

Step 3: Solve the maximization problem.(7) to obtain one specific Pareto optimal solution. Solve (7) again for each small displacement of $\Delta\mu_i$ to get trade-off rates among the memberships.

Step 4 : Stop if DM is satisfied with the obtained result. Otherwise, go to Step 5.

Step 5: Update automatically the target value and membership function parameters by using (8)-(10). Go to Step 3.

Some comments are in order here for the above algorithm. In step 4, DM is supplied with the Pareto optimal solution and the trade-off rates between the membership function. Also, in step 5, if the obtained objective value is far from the target value, it should be updated. With the new target value, DM specifies preference information (increase/decrease priority) for each objective function by considering current objective function value. Then, the new membership function parameter would be calculated by the computer.

B.5 Application to the Test System

The proposed algorithm has been applied to the IEEE 118 test system, the results of which are shown in Table 1. First, DM must solve three single-objective OPF problems to obtain maximum and minimum values of the objective functions, denoted respectively by $f_i^{\max}$ and $f_i^{\min}$, and then decide the initial membership function parameters f_{i0}, f_{i1} and the initial target value corresponding to 80% of the membership function. In this study, β is selected as 1/10 of f_{i0}.

Since the obtained solution in STAGE 1 was far from the target value, DM has decided to change it to (25.0, 0.58, 140.0). Even with the new target values, objective functions f_1 and f_2 are still unsatisfactory and need to be improved. To achieve this, the priority of f_1 and f_2 were increased and that of f_3 was decreased by (8) - (10). In STAGE 2, although f_1 and f_2 were still unsatisfactory, DM assumed that it is still possible to improve results with the same target values. Therefore, DM supplied the same target values at the next stage to further improve objective functions f_1 and f_2. The membership function parameters were updated in the same manner as in STAGE 1. After the optimization in STAGE 3, all the objective functions have satisfied the target values and thus a satisfactory solution has been derived.

Table 1 Interactive processes to the DM

STAGE	1			2			3		
Objective Function	f_1	f_2	f_3	f_1	f_2	f_3	f_1	f_2	f_3
Target Value	23.6	0.56	138.8	25.0	0.58	140.0	25.0	0.58	140.0
	22.0	0.48	136.0	22.0	0.48	136.0	22.0	0.48	136.0
	30.0	0.80	150.0	31.0	0.81	159.0	38.1	0.80	180.0
F(*x*)	26.0	0.64	138.0	25.2	0.61	139.0	25.0	0.59	140.0
m(*x*)	0.49	0.50	0.85	0.64	0.59	0.86	0.81	0.67	0.92

f_1 : Economy index, f_2 : Environment index, f_3 : Security index

F(*x*) : the value of objective function, *m*(*x*) : the degree of satisfaction

f_{i0}, f_{i1} : the membership parameter

B.6 Discussions and Concluding Remarks

Numerical simulations have been carried out on the IEEE 118 node test system to demonstrate the capability of the proposed algorithm with focusing on the execution time and iteration counts. Results are listed in Table 2, where the execution time of the ε-constrained method is also shown for the purpose of comparison. As the objective function of the proposed method with linear membership functions is the sum of the constituent objective functions, the sparsity structure of the system matrix becomes the same as that of the single objective OPF, thus giving rise to similar results as those of the conventional OPF. On the other hand, since the ε-constrained method deals with objective functions that has not been selected as the main objective as constraints, the sparsity of the Hessian matrix is disrupted, increasing the time per iteration. This property together with an increase in the iteration counts has doubled its total execution time as compared with the single objective OPF.

Table 2 Execution results for each OPF method

	Single objective OPF	Fuzzy multi-objective OPF	ε -constrained multi-objective OPF
Compose W matrix [sec]	0.20721	0.36087	0.37012
LU factorization [sec.]	1.07315	1.07738	1.42480
Enforce Inequality [sec.]	0.00684	0.00613	0.00684
Execution time / iteration [sec.]	1.28721	1.44438	1.80176
Total Time [sec.]	5.24699	5.87073	10.89917
Iteration	4	4	5

It is possible for the fuzzy coordination method to reflect the intention or preference of DM on the objectives, and moreover its convergence characteristics are roughly the same as that of the single-objective OPF. An optimal solution obtained in the additive fuzzy decision set has a guarantee to be a Pareto optimal unless membership function $\mu_i(x)$ takes on either 1 or 0, and therefore it is not necessary to make the validation test for optimality. In this section, we have shown the effectiveness of the interactive algorithm for determining a unique solution by means of fuzzy coordination. Of course, there is much room for further refinement such as a research for cases with other types of membership functions.

B.7 References

[1] A.M. Sasson, "Nonlinear programming solutions for load flow, minimum-loss and economic dispatching problems," IEEE Transactions on Power Apparatus and Systems, vol. PAS-88, No. 4, pp. 399-409, April 1969

[2] H. H. Happ, "Optimal power dispatch, " IEEE Transactions on Power Apparatus and Systems, vol. PAS-93, No. 3, pp. 820-830, May/June 1974

[3] F. Beglari, "Optimal economic-environmental generation

expansion planning, " 5th PSCC Proceedings, Paper 1.1/7, vol. 1, September 1975

[4] M.R. Gent and J.W. Lamont, "Minimum-emission dispatch, " IEEE Transactions on Power Apparatus and Systems, PAS-90, No.6, pp.2650-2660, Nov/Dec 1971

[5] J.K. Delson, "Controlled emission dispatch, " IEEE Transactions on Power Apparatus and Systems," vol.PAS-93, No. 5, pp.1359-1366, Sept./Oct. 1974

[6] F.W. Gembicki and Y.Y. Haimes, "Approach to performance and sensitivity multi-objective optimization: The goal attainment method," IEEE Transactions on Automatic Control, vol. AC-20, No. 12,pp.769-771, (Short Paper) December 1975

[7] R. Yokoyama, et al, "Multiobjective optimal generation dispatch based on probability security criteria," IEEE Transactions on Power Systems, Vol.3, No.1, pp.317-324, February 1988

[8] V. Miranda, J.T.Saraiva, "Fuzzy Modeling of Power System Optimal Power Flow", IEEE Transaction on Power Systems Vol. 7, No. 2, pp. 843-849, May 1992

[9] K. H. Abdul-Rahman, et al., "Fuzzy-set-based Linear Programming Approach to Optimal Reactive Power Control", Transaction on Power Systems, Vol. 8, No. 2 , pp. 662-670, 1993

[10] M. Sakawa, " Fuzzy Sets and Interactive Multiobjective Optimization", Plenum Press, 1993

Chapter 8 Hybrid Techniques in Fuzzy Logic

Abstract- This paper presents hybrid applications of fuzzy logic in power systems. In particular, this paper focuses on neuro fuzzy models as a hybrid model. The typical neuro fuzzy models are reviewed to understand the trend in the modeling. Next, the applications of the neuro fuzzy models are described in load forecasting, fault detection/diagnosis, system control, and modeling/analysis. Furthermore, the future direction of fuzzy logic is mentioned.

Keywords: fuzzy logic, hybrid applications, neuro fuzzy

A. Introduction

It is expected that intelligent systems allows to smooth power system operation and planning. In practice, it is not easy to understand and control power systems appropriately. Power systems have the following complicated factors:

- nonlinear dynamics
- periodicity and/or randomness
- large-scale
- discrete event systems, etc.

They make power systems more complex so that power system operators have difficulty in carrying out on-line computation. Aside from the analytical methods, the intelligent systems aims at solving the following:

1) Problems without any analytical algorithms
2) Problems that may be expressed by knowledge and experiences
3) Pattern recognition problems in which the nonlinear relationship between input and output variables are identified

Among the intelligent systems, artificial neural networks(ANNs) inspired by the biological nerve system have been developed to carry out the distributed information processing. They consist of a group of units called "neurons" that are analogous to nerve neurons. The multi-perceptron(MLP) is the mainstream of ANNs due to the universal nonlinear approximator. According to the supervised learning, the weights between neurons are optimized to obtain a good model. However, it is pointed out that MLP uses the black-box like description of the inference process so that the relationship between input and output variables is not clear. As a result, it is hard to capture the tendency of the predicted value as well as cause and effect.

Table 1 Comparison between MLP and Fuzzy Inference

Features	MLP	Fuzzy
Capability of Approximating Any Functions	X	X
Model Determination through Learning	X	
Easiness of Inference Process		X

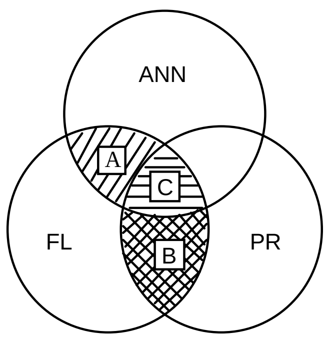
On the other hand, fuzzy inference is also one of promising intelligent system approaches. Kosko made a proof that fuzzy inference allows us to approximate any nonlinear functions with finite fuzzy rules[6]. Fuzzy inference has the same ability as MLP in approximating any functions. The advantage of the fuzzy inference over ANN is to capture cause and effect in the inference process. Due to the clear relationship between input and output variables, the results are intuitively more understandable and user-friendly. However, the conventional fuzzy inference does not have the learning function like MLP. It means that it is questionable whether the model reflects the obtained data.Table 1 shows a comparison between MLP and fuzzy inference. Therefore, models are required to possess the advantages of both MLP and fuzzy inference.

B. Soft Computing

This section briefly describes soft computing (SC) to understand the concept of the hybrid methods with fuzzy[3]. Zadeh proposed SC in which the human-like approximate ability without high precision is simulated to solve the complex problems. It consists of the following:

- Fuzzy logic (FL)
- Artificial neural network (ANN)
- Probabilistic reasoning(PR)

Fig. 1 shows the concept of SC. The objective is to achieve tracterability, robustness, and low solution cost of the systems to be studied for solving complicated problems. It is different from the conventional(hard) computing in a



Note) FL: Fuzzy Logic
ANN: Artificial Neural Network
PR: Probabilistic Reasoning

Fig. 1 Concept of Soft Computing

sense that it is tolerant of imprecision, uncertainty, and partial truth. It should be noted that SC implies a discipline that allows three methods to work out the problem in a complementary way. Fuzzy inference with fuzzy rules is suitable for identifying the nonlinear relationship between input and output variables with high accuracy. ANN works as a mathematical model that simulated the right-side of human brain or nerve systems It should be noted that PR corresponds to the following:

- Belief Networks[4]
- Genetic Algorithms (GA)[5]
- Chaos Systems[6]
- Learning Theory, etc.

Belief networks are a means of representing uncertain knowledge from experts. They show the probabilistic dependency among a set of variables. They are called Baysian networks, knowledge maps, or qualitative probabilistic networks. GA is a heuristic probabilistic optimization technique that has been inspired by the natural selection. The algorithm includes genetic operators such as crossover, mutation, reproduction, etc. so that better solutions are evaluated. Unlike the conventional methods, it is expected that GA gives solutions near a global optimum. Chaos systems are used in temporal pattern search in nonlinear optical resonator, deterministic nonlinear prediction of economics, etc.

FL, ANN, and PR cooperate with each other to handle uncertain information that is not expressed b y crisp numbers. In other words, SC allows a computer to behave like a human and solve complex problems. Specifically, FL, ANN, and PR are related to imprecision, learning, and uncertainty, respectively. The three approaches overlap with each other as shown in Fig. 1. Areas A, B, and C in the figure are cooperative rather than competitive. Suppose that SC is applied to a problem. The role of FL, ANN, and PR depends on the problem. The combination results in an advantageous method. Focusing on the role of FL, most of the approaches correspond to Areas A and B in power systems. In particular, Area A is called neuro fuzzy. In the next section, typical neuro fuzzy models are outlined from a standpoint of the role of fuzzy.

C. Typical Neuro-Fuzzy Models

This section briefly introduces typical neuro fuzzy models[7] as one of promising intelligent systems although the integration of fuzzy with other technologies is found[8]. The degree of the integration of FL with ANN becomes higher as the model proceeds to Type K from Type A in neuro fuzzy models. Namely, Type A implies the most primitive neuro fuzzy model.

C.1 Type A

Suppose that a system has two functions of fuzzy rules and ANN independently. The fuzzy rules handle some input and output variables while ANN does the others (see Fig. 2). It can be seen that the fuzzy rules deal with the different input variables from those of ANN. The model is referred to as Type A. Fuzzy rules are used for the problem in which the knowledge and experience of experts are described. On the other hand, ANN is used for the problem that fuzzy rules can not handle. Thus, there is no relationship between fuzzy rules and ANN.

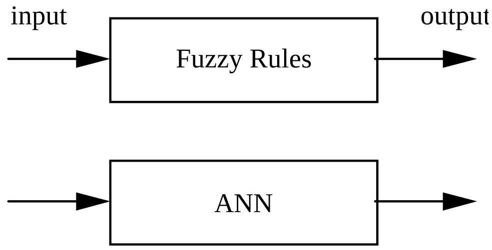
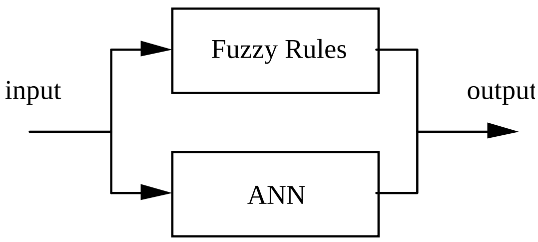
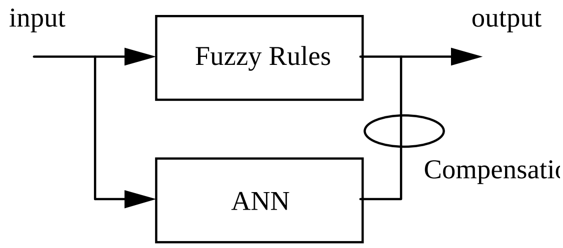


Fig. 2 Neuro Fuzzy Type A



(a) Unified Model

Fig. 3 Neuro Fuzzy Type B



(b) Compensation Model

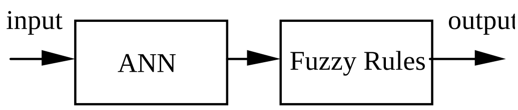


Fig. 4 Neuro Fuzzy Type C

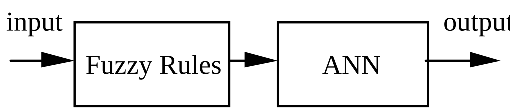


Fig. 5 Neuro Fuzzy Type D

C.2 Type B

Fuzzy rules and ANN may be placed in parallel as shown in Fig. 3. The model is called Type B. Depending on the role of FL and ANN, it may be divided into the following model:

1) Unified model(see Fig. 3(a))
2) Compensation model(see Fig. 3(b))

In the unified model, information processing is equally done for FL and ANN. Also, FL compensates the results obtained by ANN in the compensation model, and vice verse.

C.3 Types C & D

FL and ANN may be placed in series so that two-phase inference is possible if ANN may be regarded as a kind of inference. Regarding the two-phase inference, two schemes are allowed to evaluate output variables as follows:

a) ANN plus Fuzzy rules(see Fig. 4)
b) Fuzzy rules plus ANN(see Fig. 5)

The former is called Type C while the latter is Type D. The choice of the types depends on the problems .

C.4 Type E

Fig. 6 shows Type E of the neuro fuzzy model. A fuzzy model is used to handle fuzzy rules in which the goal and parameters of the fuzzy control are evaluated. It should be noted that ANN contributes to determination of constructing the fuzzy rules. In other words, ANN plays an assistant role in the fuzzy model.

C.5 Type F

The model called Type F is based on the ANN model that makes use of fuzzy rules in determining the weights between neurons (see Fig. 7). The parameters of ANN such as the weights between neurons and the threshold value of neurons is evaluated by the fuzzy rules. That allows to speed up the ANN learning and reduce the model errors. The role of ANN and fuzzy rules in this model corresponds to that of fuzzy and ANN of Type E , respectively.

C.6 Type G

Type G makes use of the integration of fuzzy rules and ANN so that the supervised learning of ANN is used to evaluate the membership function shape and the weight of true value of fuzzy rules (see Fig. 8). As a learning scheme, the steepest decent method is used the error backpropagation algorithm of the multilayer perceptron. The difference between Types E and G is that only the function of the ANN learning is used in Type G to tune up fuzzy rules to improve the solution accuracy.

C.7 Type H

The model of Type H has function that fuzzy rules of if-then are expressed with the ANN construction. The model is useful in a sense that the computation process of fuzzy inference or fuzzy control may be represented by a learning model. This concept is shown in Fig. 9. Since the ANN represents the fuzzy rules, the output variable after the ANN learning corresponds to the inference value of fuzzy model.

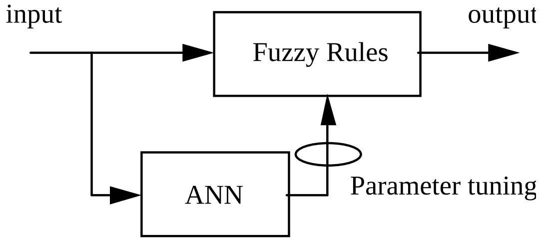


Fig. 6 Neuro Fuzzy Type E

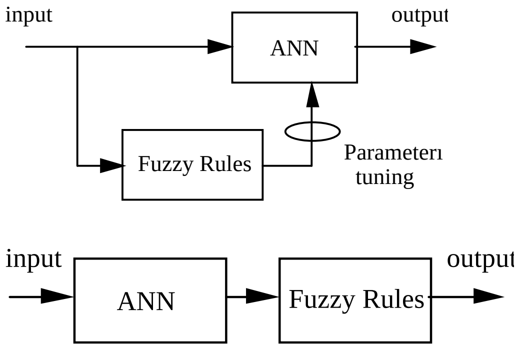


Fig. 7 Neuro Fuzzy Type F

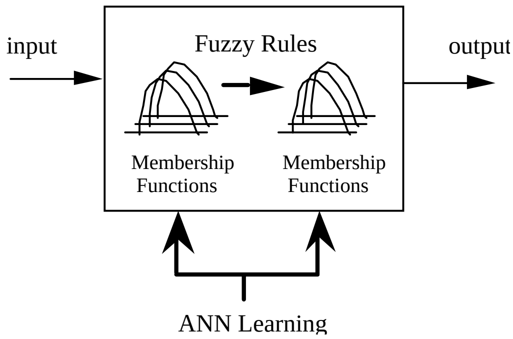


Fig. 8 Neuro Fuzzy Type G

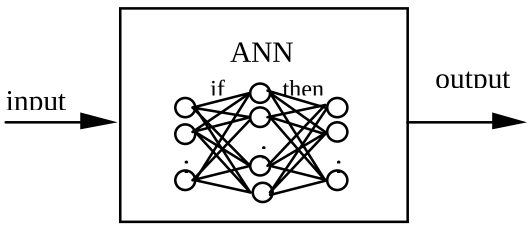


Fig. 9 Neuro Fuzzy Type H

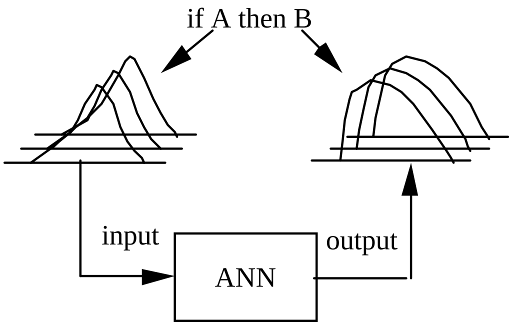


Fig. 10 Neuro Fuzzy Type I

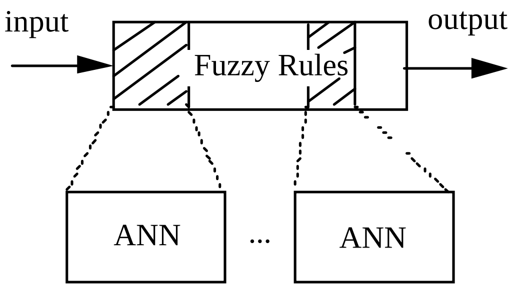


Fig. 11 Neuro Fuzzy Type J

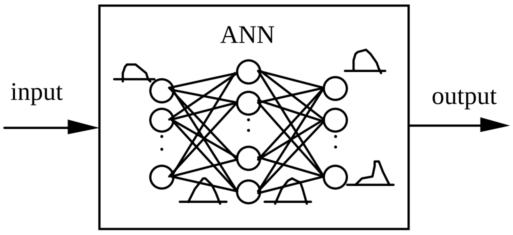


Fig. 12 Neuro Fuzzy Type K

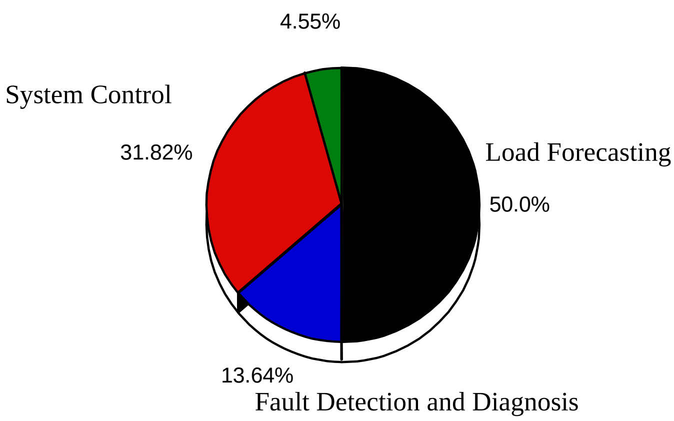


Fig. 13 Main Specific Problems

C.8 Type I

A fuzzy inference model is identified with ANN to clarify the relationship between the premise and consequence of fuzzy rules as shown in Fig. 10. The ANN model is constructed after fuzzy sets of the premise and consequence are assigned to input and output of the learning data of ANN, respectively. As a result, the input and output variables of the model correspond to the value of the fuzzy membership functions. Specifically, studies on ANNs representing fuzzy rules and fuzzy operators have been done.

C.9 Type J

A part of fuzzy rules in the fuzzy model is expressed by ANNs in Type J (see Fig. 11). The ANN model is used to substitute for some fuzzy rules so that the errors of the fuzzy membership functions or the consequence are reduced. The difference between Types G and J is that ANNs becomes a subsystem of fuzzy rules in Type J .

C.10 Type K

The model corresponds to a generalized neuro fuzzy model. It is a kind of an extension of ANN in a sense that the weights between neurons are fuzzified. That implies that it can handle input data as a fuzzy number. It is necessary to develop more sophisticated learning algorithms in consideration of fuzzy logic.

D. Typical Applications of Neuro-Fuzzy Models to Power Systems

This section reviews typical applications of neuro fuzzy models to power systems. Fig. 13 gives an overview of main specific problems with neuro fuzzy models. The areas may be listed as follows:

-Load forecasting (50.0%)
-Fault Detection and Diagnosis(13.64%)
-System Control(31.82%)
-Analysis and Modeling(4.55%)

In load forecasting, a neuro fuzzy model is used as one of tools to deals with time series analysis of a load. Among the load forecasting problems, short-term load forecasting is of main concern. High accuracy of the load forecasting improves security and generation cost. However, the forecasting problem is not so easy due to the complicated factors such as nonlinearity, weather conditions, etc. The neuro fuzzy models allow us to carry out adaptive forecasting efficiently.

Table 2 Typical Applications of Neuro Fuzzy Models to Power Systems

Areas	References	Neuro Fuzzy Type	ANN Model	Problems to be studied
Load Forecasting	[LF1]	A	MLP	One-day ahead prediction
	[LF2]	C	MLP	1-48hours ahead prediction
	[LF3]	C	MLP	same as [LF2]
	[LF4]	G	MLP	One-hour ahead prediction
	[LF5]	G	MLP	One-day ahead prediction
	[LF6]	D	MLP	One-day ahead prediction
	[LF7]	G	MLP	One-day ahead prediction
	[LF8]	D	MLP	One-day ahead prediction
	[LF9]	C	MLP	One-day ahead prediction
	[LF10]	D	MLP	One-day ahead prediction
	[LF11]	G	MLP	Optimal structure of MFs
Fault Detection /Diagnosis	[FD1]	D	MLP	Animal fault detection
	[FD2]	D	MLP	Shorted turns in widingings
	[FD3]	D	MLP	Equipment conditions
System Control	[CN1]	C	MLP	Hybrid PSS
	[CN2]	C	MLP	Hybrid PSS
	[CN3]	C	MLP	Hybrid PSS
	[CN4]	C	MLP	Hybrid controller
	[CN5]	D	MLP	Extinction angle control
	[CN6]	G	MLP	Excitation controller
	[CN7]	C	MLP	PWM controller for induction machine
Analysis/Modeling	[AM1]	[illegible]	MLP	Load modeling

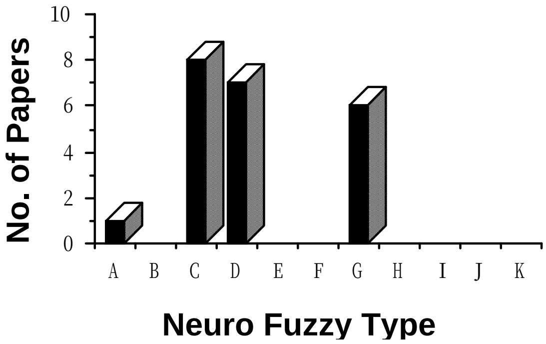


Fig. 14 Classification of Papers on Neuro Fuzzy Types

Table 3 Expected Functions of Emerging Technologies

Technologies	Expected Functions
ANN	Neuro Fuzzy
Metaheuristics	Optimal Fuzzy Structure
Expert Systems	Fuzzy expert systems
Multi-Agent Syst.	Fuzzy Modules
A-life	Evolution of Structure
Chaos	Chaos model with Fuzzy Logic

Fault detection/diagnosis is one of challenging problems in power systems. The neuro fuzzy models -identifies the type and location of faults with a given set of power system conditions, measurements, alarms, etc. Through a given set of input variables, the neuro fuzzy model handles selecting solution candidates. However, the complexity increases significantly as the system size increases. As a result, it is still questionable whether the simple neuro fuzzy model gives the "true" solution in large scale systems.

System control tries to construct a control method in power systems that is based on a kind of pattern recognition rather than optimal control theory. The method allows to carry out on-line control although strictly speaking, it does not give the optimal solution in terms of control theory.

Going into some detail, Table 2 shows an detailed overview of papers in terms of the neuro fuzzy type, the used ANN model, and the problems to be studied. The following can be observed:

• Load forecasting is the most popular area in neuro fuzzy models. That is because load forecasting does not need a lot of the input variables while it gives a single output variable such one-step ahead prediction. In other words, the problem is less difficult than other problems.

• Neuro fuzzy models C and D are widely used for simplicity although they are less sophisticated than other models E-K(see Fig. 14). It can be seen that the model with the high degree of the integration of fuzzy logic and ANN have not been sufficiently studied in power systems.

• The used ANN model is MLP in all the cases. MLP is easier to incorporate fuzzy logic into the model structure.

This paragraph describes future work that enhances the performance of fuzzy logic. It is a natural research direction to make use of other emerging technologies to overcome drawbacks of fuzzy logic. Table 3 gives the expected functions of emerging technologies for fuzzy logic. In Fuzzy-ANN, the membership function is learned through the BP algorithm. Also, metaheuristics is used to find out a solution near a global minimum in determining optimal structure of the fuzzy membership functions. Ref. [LF11] handles SA based learning for constructing the optimal fuzzy membership functions .

E. Conclusions

This paper has provided an overview on hybrid models of fuzzy logic in power systems. In particular, this paper has described neuro fuzzy models that is the integration of fuzzy logic with artificial neural networks. As the application areas, load forecasting, fault detection/diagnosis, system control , and analysis/modeling were of main concern although there exist a variety of application areas. In addition, the integration of fuzzy logic with other emerging technologies was described as future work.

F. References

[1] B. Kosko, "Neural Networks and Fuzzy Systems" Prentice Hall, Englewood Cliffs, NJ, U.S.A., 1992.

[2] B. Kosko, "Fuzzy Systems as Universal Approximators," IEEE International Conference of Fuzzy Systems 1992, pp. 1153-1162, San Diego, California, U.S.A., Mar. 1992.

[3] F. Aminzadeh and M. Jamshidi (Eds.), "Soft Computing" Prentice Hall, Englewood Cliffs, NJ, U.S.A., 1994.

[4] R.D. Schechter, et al., "Symbolic Probabilistic Inference in Belief Networks" AAAI-90, 1, pp. 126-131, 1990.

[5] D.E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison Wesley Publishing Company 1989.

[6] K. Aihara and R. Tokunaga(Eds.), "Applied Strategies of Chaos," Ohm-sha, Tokyo, Japan 1993(in Japanese).

[7] I. Hayashi, and T. Furuhashi (Eds.), "Fuzzy Neural Networks, " Asakura-Shoten, Tokyo, Japan, 1996(in Japanese).

[8] A.K. David and Z. Ronga, "An Expert System with Fuzzy Sets for Optimal Planning," IEEE Trans. on Power Systems, Vol. 6, No. 1, pp. 59-65, Feb. 1991.

Load Forecasting

[LF1] K.H. Kim, et al., "A Hybrid Model of Artificial Neural Network and Fuzzy Expert System for Short-Term Load Forecasting," Proc. of ESAP'93, pp. 164-168, Melbourne, Australia, Jan. 1993.

[LF2] P.K. Dash, et al., "A Hybrid Artificial Neural Network-Fuzzy Expert System for Short Term Load Forecasting,"

Proc. of ESAP'93, pp. 175-180, Melbourne, Australia, Jan. 1993.
[LF3] P.K. Dash, et al., "A Fuzzy Adaptive Correction Scheme for Short Term Load Forecasting Using Fuzzy Layered Neural Network," Proc. of ANNPS'93, pp. 432-440, Yokohama, Japan, Apr. 1993.
[LF4] H. Mori and H. Kobayashi, "A Fuzzy Neural Net for Short-term Load Forecasting," Proc. of ISAP'94, pp. 775-782, Montpellier, France, Sep. 1994.
[LF5] P.K. Dash, et al., "A Comparative Study of Load Forecasting Models Using Fuzzy Neural Networks," Proc. of ISAP'94, pp. 865-874, Montpellier, France, Sep. 1994.
[LF6] D. Srinivasan, et al., "Forecasting Daily Load Curves Using a Hybrid Fuzzy-Neural Approach," IEE-Proc. C, Vol. 141, No. 6, pp. 561-567, Nov. 1994
[LF7] A.G. Bakirtzis, et al., "Short Term Load Forecasting Using Fuzzy Neural Networks," IEEE PES 1995 Winter Meeting, Paper No. 95 WM155-2 PWRS, New York, NY, Jan. 1995.
[LF8] D. Srinivasan, et al., "Demand Forecasting Using Fuzzy Neural Computation with Special Emphasis on Weekend and Public Holiday Forecasting," IEEE PES 1995 Winter Meeting, Paper No. 95 WM158-6 PWRS, New York, NY, Jan. 1995.
[LF9] K.-Ho Kim, et al., "Implementation of Hybrid Short-Term Load Forecasting System Using Artificial Neural Networks and Fuzzy Expert Systems," IEEE PES 1995 Winter Meeting, Paper No. 95 WM159-4 PWRS, New York, NY, Jan. 1995.
[LF10] P. K. Dash, et al., "Peak Load Forecasting Using a Fuzzy Neural 1," Electric Power Systems Research, Vol. 32, No.1, pp. 19-23, Jan. 1995.
[LF11] H. Mori, and H. Kobayashi, "Optimal Fuzzy Inference for Short-term Load Forecasting," IEEE Trans. on Power Systems, Vol. 11, No. 1, pp. 390-396, Feb. 1996.

Fault Detection /Diagnosis

[FD1] M. -Y. Chow, et al., "Neural-Fuzzy Hybrid System for Distribution Fault Causes Identification," Proc. of ANNPS'93, pp. 427-431, Yokohama, Japan, Apr. 1993.
[FD2] M.A. El-Sharkawi, et al., "Neural Network Application to Short Turn Localization Using Fuzzified Data," Proc. of ISAP'94, pp. 129-134, Montpellier, France, Sep. 1994.
[FD3] K. Tomsovic and A. Amar, "On Refining Equipment Condition Monitoring Using Fuzzy Sets and Artificial Neural Nets," Proc. of ISAP'94, pp. 363-370, Montpellier, France, Sep. 1994.

System Control

[CN1] K. Saitoh and S. Iwamoto, "Application of Neural Network Based Fuzzy Control to Power System Generator," Proc. of ANNPS'91, pp. 144-148, Seattle,WA, U.S.A., July 1991.
[CN2] T. Hiyama, "Application of Neural Network to Real Time Tuning of Fuzzy Logic PSS," Proc. of ANNPS'93, pp. 421-426, Yokohama, Japan, Apr. 1993.
[CN3] A.M. Sharaf and T.T. Lie, "A Neuro-Fuzzy Hybrid Power System Stabilizer," Electric Power Systems Research, Vol. 30, No. 1, pp. 17-23, Jun. 1994.
[CN4] H.-C. Chang and M.-H. Wang, "Neural Network Based Self-Organizing Fuzzy Controller for Transient Stability of Multimachine Power Systems," IEEE PES 1994 Summer Meeting, Paper No. 94 SM372-3 EC, San Francisco, CA, Jul. 1994.
[CN5] R. Jayakrishna, et al., "Neuro-Fuzzy Controller for Enhancing the Performance of Extinction Angle Control of Inverters in a MTDC-AC System," Proc. of ANNPS'93, pp. 206-211, Yokohama, Japan, Apr. 1993.
[CN6] S. Tyc and V. Quintana, "Linear Optimal Regulator Based on a Fuzzy Logic Neural Network," Proc. of ISAP'94, pp. 63-70, Montpellier, France, Sep. 1994.
[CN7] K.-B. Cho, "A Neuro Fuzzy Controller for Inverter Fed Variable Speed Induction Motor Drive on the Power System," Proc. of ANNPS'93, pp. 47-52, Yokohama, Japan, Apr. 1993.

Analysis/Modeling

[AM1] H. Mori, "Fuzzy Inference Based Identification of Load Characteristics," IEEE Proc. of ISCAS'96, Vol. 1, pp. 621-624, Atlanta, GA, May 1996.

G. Appendix

References on Applications of Artificial Neural Networks to Power Systems (From Jan. 1989 to Jul. 1995)

・IEEE Transactions/Journals/Proceedings
- PES Trans. on Power Systems
- PES Trans. on Power Delivery
- PES Winter/Summer Meeting
- Power Industry Computer Application Conf.(PICA)
- Int'l Symp. on Circuits and Systems(ISCAS)

・Other Journals/Proceedings
-ANNPS(International Forum on Application of Neural Networks to Power Systems; Seattle, WA, '91, and Yokohama, Japan, '93)
-Electrical Power Systems Research
-ESAP(International Symp. on Expert Systems Application to Power Systems; Melbourne'93)
-IEE-Proc. Pt. C
-ISAP(International Conference on Intelligent System Application to Power Systems; Montpellier, France, '94)