

Heuristic Constraints Enforcement for Training of and Knowledge Extraction from a Fuzzy/Neural Architecture—Part I: Foundation

Mo-Yuen Chow, *Senior Member, IEEE*, Sinan Altug, *Student Member, IEEE*, and H. Joel Trussell, *Fellow, IEEE*

Abstract—Using fuzzy/neural architectures to extract heuristic information from systems has received increasing attention. A number of fuzzy/neural architectures and knowledge extraction methods have been proposed. Knowledge extraction from systems where the existing knowledge limited is a difficult task. One of the reasons is that there is no ideal rulebase, which can be used to validate the extracted rules. In most of the cases, using output error measures to validate extracted rules is not sufficient as extracted knowledge may not make heuristic sense, even if the output error may meet the specified criteria. This paper proposes a novel method for enforcing heuristic constraints on membership functions for rule extraction from a fuzzy/neural architecture. The proposed method not only ensures that the final membership functions conform to *a priori* heuristic knowledge, but also reduces the domain of search of the training and improves convergence speed. Although the method is described on a specific fuzzy/neural architecture, it is applicable to other realizations, including adaptive or static fuzzy inference systems. The foundations of the proposed method are given here in Part I. The techniques for implementation and integration into the training are given in Part II, together with applications.

Index Terms—Constraint enforcement, knowledge extraction, neural/fuzzy architectures, set theory.

I. INTRODUCTION

USING fuzzy/neural architectures to extract heuristic information from systems has received increasing attention recently. For this purpose, fuzzy/neural architectures and rule extraction methods have been analyzed [1]–[17]. Methods proposed for fuzzy information extraction include methods based on clustering in input, output, or input/output product spaces; methods based on covariance of fuzzy partitions; and methods based on direct matching of the input/output data [17]–[19].

In the process of knowledge extraction, even for cases where we assume no *a priori* information about the rulebase exists, information about the extracted knowledge that may be useful in evaluating whether the rulebase makes heuristic sense may still be available. In this paper, a method of incorporating such information into the training and rule extraction procedure for a fuzzy/neural system is proposed. This method is called the *heuristic constraints enforcement* method.

Manuscript received November 11, 1997; revised September 14, 1998. This work was supported by the National Science Foundation under Grant ECS-9521 609.

The authors are with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695 USA.

Publisher Item Identifier S 1063-6706(99)03420-7.

A. Problem Definition

The aim of using a fuzzy/neural architecture (FZ/NN) is to find, through training, a fuzzy inference system (FIS) that represents the underlying system while also satisfying an error criterion. This fuzzy inference system can be visualized as being composed of: 1) rules; 2) fuzzy labels (or sets) used in these rules; and 3) fuzzy inference system specifications (e.g., method of intersection). This conceptual fuzzy inference system is implemented using an architecture equivalent to a neural network called the fuzzy/neural architecture. Because the specifications of this architecture are fixed in terms of fuzzy inference operations (intersection, aggregation, defuzzification, etc.), the underlying fuzzy inference system specifications are also fixed. The part that is allowed to change during the training of the FZ/NN architecture is the connection weights associated with the rules and the membership functions corresponding to the fuzzy sets utilized in this rulebase. Hence, the problem of finding an FIS can be equivalently interpreted as determining the relative importance of fuzzy rules and the membership functions corresponding to the fuzzy sets in these rules.

Of these, the fuzzy rules of an FIS can be represented in the FZ/NN architecture by a layer of connection weights. This representation has extra degrees of freedom compared to a *conventional* FIS, as firing strengths can be assigned to each rule through the continuous values of the above mentioned connection weights. The change of the structure of fuzzy sets is via adapting the membership functions corresponding to the sets, which are represented by *parameters* in the FZ/NN. Consequently, the conceptual problem of finding membership functions and rules of the FIS is, in fact, finding parameters that represent the membership functions and the relative importance of the rules of the FIS, respectively.

B. Set Theoretic Framework

Set theoretic estimation has been used in a wide range of research areas and has proven to be an effective tool to incorporate information into numerous problems [20]–[23]. In the set theoretic formulation of a problem, every piece of information, including information about the system, the solution and external factors are represented with sets. The intersection of all these sets form the set or family of acceptable solutions, referred as the *feasibility set* [20]. Therefore, in set theoretic estimation, the optimal solution is not sought, rather, a set of solutions is defined as a class of objects that is consistent

with all the information. The incorporated information includes information acquired from the observed data as well as *a priori* information [23]. One of the most powerful features of set theoretic formulation is the fact that the “solution space” may be in various different forms depending on the object that is being estimated [23].

Using principles of set theoretic estimation, a methodology for enforcing heuristic constraints on membership functions for information extraction from a fuzzy/neural architecture is developed in this paper. Moreover, a procedure for integrating these constraints into the training of the FZ/NN is also proposed, so that the extracted membership functions are guaranteed to conform to the heuristic knowledge that is embodied into the formulation in the form of constraint sets. Also, with this method, the domain of search in FZ/NN training is reduced as the search is carried out only in the region of the space where the constraints are satisfied, improving convergence.

This paper focuses on the novel approach of incorporating *a priori* information as constraints so that after the successful training of the FZ/NN system, the system can provide both accurate input–output mapping and the semantics/qualitative reasoning behind the input–output mapping. The training can also be thought as an optimization problem under constraints. Many techniques can be used to solve optimization problems under constraints after the problem under investigation has been formulated by the proposed method. Since optimization techniques and computational advantages of FZ/NN system are not the focus of this paper, we chose *an* optimization technique that can provide a satisfactory solution in order to demonstrate the FZ/NN approach. The advantages of the set theoretic based optimization method are the following.

1. It is easy to directly implement the heuristic constraints, i.e., program the projections to the constraint sets.
2. It is easy to interface the method with the existing fuzzy/neural programs.
3. The constraints are guaranteed to be satisfied, as opposed to penalty function methods.
4. Convergence is guaranteed since the projections are nonexpansive mappings.

The process of incorporating heuristic information include the following steps:

- 1) defining heuristic constraints using *a priori* information;
- 2) forming constraint sets corresponding to the constraints;
- 3) enforcing constraints on the membership functions to obtain solutions for membership functions from the acceptable solutions set;
- 4) incorporating the enforcement method into the training of the FZ/NN.

The general definitions related to the posed problem are given in Section II. The fuzzy/neural architecture and the training algorithm are briefly presented in Section III. The methodology for enforcing heuristic constraints on membership functions is given in Section IV following the above steps. The procedure for implementation and incorporation of the method into the training algorithm of the fuzzy/neural architecture, as well as experiments and conclusions are given in Part II of this paper [24].

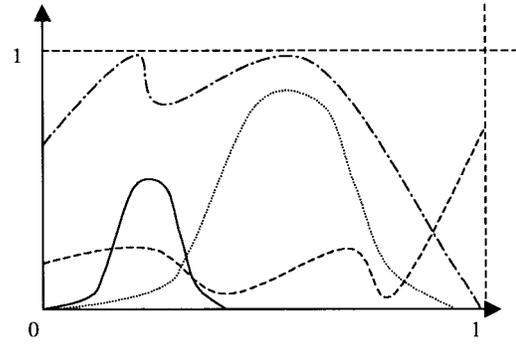


Fig. 1. Examples of elements of Ω_{m_i} .

II. DEFINITIONS

This section gives the general definitions used in this paper. Part II also follows the same notation.

A. General Definitions

The inputs of the fuzzy/neural (FZ/NN) structure are defined on the *input space* $\Omega_{\mathbf{x}}$, which is formed by the Cartesian product of n spaces

$$\Omega_{\mathbf{x}} = \Omega_{x_1} \times \Omega_{x_2} \times \cdots \times \Omega_{x_n} \quad (1)$$

n is the number of inputs. The *output space* $\Omega_{\mathbf{y}}$ is where the output of the FZ/NN is defined

$$\Omega_{\mathbf{y}} = \Omega_{y_1} \times \Omega_{y_2} \times \cdots \times \Omega_{y_m} \quad (2)$$

where m is the number of outputs. Thus, the input vector is $\mathbf{x} = [x_1, \dots, x_n]^T$ and the corresponding output vector is $\mathbf{y} = [y_1, \dots, y_m]^T$. Two types of parameters for the FZ/NN architecture are defined—the *membership function parameters*, which are used to parameterize the membership functions, and the *rule parameters*, which are associated with the relative importances of the rules. We define

$$\Omega_{\mathbf{p}} = \Omega_{p_1} \times \Omega_{p_2} \times \cdots \times \Omega_{p_k} \quad (3)$$

as the *membership function parameter space*, (or *parameter space* in short) and

$$\Omega_{\mathbf{q}} = \Omega_{q_1} \times \Omega_{q_2} \times \cdots \times \Omega_{q_l} \quad (4)$$

as the *rule parameter space*. The numbers k and l above depend on the number of membership functions defined on each space and the number of parameters used in parameterization of each membership function.

Finally, *membership function space* $\Omega_{\mathbf{m}}$ is defined as the Cartesian product of $n + m$ spaces, where n is the number of inputs and m is the number of outputs

$$\Omega_{\mathbf{m}} = \Omega_{m_1} \times \Omega_{m_2} \times \cdots \times \Omega_{m_{(n+m)}}. \quad (5)$$

We define each Ω_{m_i} as a subset of $\mathbf{C}_{[0,1]}$, the space of continuous functions defined on the interval $[0, 1]$ and take values from $[0, 1]$

$$\Omega_{m_i} = \{\mu_i \mid \mu_i \in \mathbf{C}_{[0,1]}; \mu_i : [0, 1] \rightarrow [0, 1]\} \quad (6)$$

where $1 \leq i \leq n + m$. Fig. 1 represents several general forms of the elements of Ω_{m_i} , each of which is a function defined on $[0, 1]$.

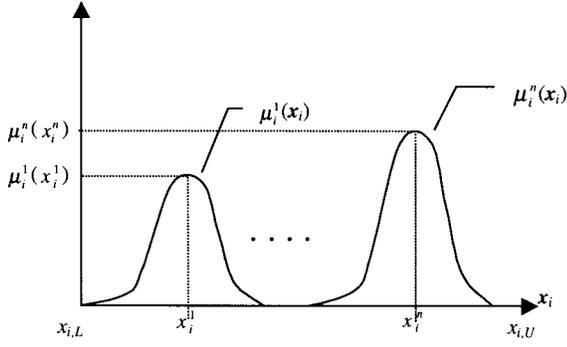


Fig. 2. The fuzzy sets defined on the membership function space Ω_{m_i} .

B. Fuzzy Set Definitions on the Membership Function Space $\Omega_{\mathbf{m}}$

This section will focus on membership functions corresponding to fuzzy sets defined on a single variable, x_i . The definitions in this section are valid for each of the input spaces $\Omega_{\mathbf{x}_i}$ and the output spaces $\Omega_{\mathbf{y}_i}$. $\Omega_{\mathbf{x}_i}$ is defined as the universe of discourse of the variable x_i

$$\Omega_{\mathbf{x}_i} = \{x_i \mid x_{i,L} \leq x_i \leq x_{i,U}\} \quad (7)$$

where $x_{i,L}$ is the lower bound and $x_{i,U}$ is the upper bound for x_i in $\Omega_{\mathbf{x}_i}$. The input and output data sets are normalized to $[0, 1]$, by linearly mapping the data to $[0, 1]$, which yields $x_{i,L} = 0$, $x_{i,U} = 1$. This normalization is done for simplification in notation, yet is not a requirement. A total of M_i fuzzy sets are defined on $\Omega_{\mathbf{x}_i}$ and labeled as X_i^j ; $j = 1, \dots, M_i$. Each X_i^j has a corresponding membership function μ_i^j , where

$$\mu_i^j : \Omega_{\mathbf{x}_i} \rightarrow [0, 1], \quad \text{for } j = 1, \dots, M_i. \quad (8)$$

For notational ease, J is defined $J = \{1, 2, \dots, M_i\}$. For each membership function μ_i^j , $j \in J$, a point where the membership function attains its maximum are labeled as x_i^j

$$\mu_i^j(x_i^j) = \max_{x_i} \{\mu_i^j(x_i)\}, \quad \forall j \in J. \quad (9)$$

There may be more than one such point in each universe of discourse. In this case, x_i^j is the smallest of these points. An example group of membership functions is given in Fig. 2.

III. THE FUZZY/NEURAL ARCHITECTURE

A. Architectural Specifications

The FZ/NN architecture is equivalent to a neural network, with layers corresponding to the fuzzy inference engine in terms of intersection, aggregation, defuzzification, and decision-making, and called shortly as FALCON in [11]. For detailed information on the FZ/NN architecture, the reader should consult [11].

Different types of membership functions can be used with the proposed methodology. Gaussian type membership functions are used in this paper because they are smooth unimodal functions which correspond well with heuristic fuzzy membership function and the parameterized mathematical form aids computation and programming. The Gaussian type of

membership functions used for the fuzzy sets defined in both input and output spaces are in the form

$$\mu_i^j(\mathbf{x}_i, a, b) = c e^{\frac{-(x_i - a)^2}{2b^2}} \quad (10)$$

where the parameters a and b represent the mean and standard deviation of the function, respectively, and c is the scaling factor which is assumed to be equal to one. Therefore, each membership function μ_i^j in $\Omega_{\mathbf{m}_i}$ can be represented with the parameter pair (a, b) in the parameter vector \mathbf{p} . The relative importance of fuzzy rules are represented by a layer of connection weights in the FZ/NN, indicated as \mathbf{q} . The FZ/NN can be visualized as a mapping $F: \Omega_{\mathbf{x}} \times \Omega_{\mathbf{p}} \times \Omega_{\mathbf{q}} \rightarrow \Omega_{\mathbf{y}}$, which maps the input \mathbf{x} to $\hat{\mathbf{y}}$, the estimate of the output \mathbf{y} , using the membership function parameters \mathbf{p} and rule parameters \mathbf{q}

$$\hat{\mathbf{y}} = F(\mathbf{x}, \mathbf{p}, \mathbf{q}). \quad (11)$$

For any input \mathbf{x} , the fuzzy/neural output estimate $\hat{\mathbf{y}}$ is obtained using (\mathbf{p}, \mathbf{q}) through (11).

The training is composed of two main parts. Following initialization of the membership functions (representing fuzzy partitions) and connection weights (representing rules), a modified competitive learning algorithm is applied to extract the rule parameters \mathbf{q} . The second part of the training is an optimal adjustment of the membership function parameters \mathbf{p} , with an effort to minimize the error measure defined on the output. A gradient descent algorithm is used to update both antecedent and consequence membership function parameters. For details the reader should refer to [11], [18].

B. Relationship of FZ/NN Architecture and FIS

The FZ/NN architecture mimics a fuzzy inference system, yet the FZ/NN has extra degrees of freedom through continuous values of \mathbf{q} used to represent the rule structures. Any FIS can be represented through the FZ/NN. Higher values of \mathbf{p} for FZ/NN indicate the relative importance of the corresponding rule in the underlying rulebase, whereas in a conventional rulebase, where each rule is considered to have equal importance or $q_i = 1$, $1 \leq i \leq l$. Fig. 3 presents the relationship between the utilized FZ/NN architecture and a conventional fuzzy inference system for a case of two inputs and one output, with two fuzzy sets defined on each input and the output.

The mathematical interpretations of the computations at points labeled from one to six in Fig. 3(a) and (b) are given in Table I. In the table, $u_i^{(j)}$ and $o_i^{(j)}$ are the i th input and the i th output of the j th layer of the FZ/NN, respectively. Furthermore, m and σ represent the membership function parameters and the layer four weights $w_{ij}^{(4)}$ represent the rule parameters.

As mentioned previously, the training of the FZ/NN is equivalent to adapting parameters $\mathbf{p} \in \Omega_{\mathbf{p}}$ and $\mathbf{q} \in \Omega_{\mathbf{q}}$. These parameters are then used to implement membership functions and the connection weights corresponding to rule structures in the FZ/NN. The conceptual diagram of the FZ/NN architecture is depicted in Fig. 4 where $\Omega_{\mathbf{x}}$, $\Omega_{\mathbf{y}}$, $\Omega_{\mathbf{m}}$, $\Omega_{\mathbf{p}}$, and $\Omega_{\mathbf{q}}$ represent the topographical locations of the elements

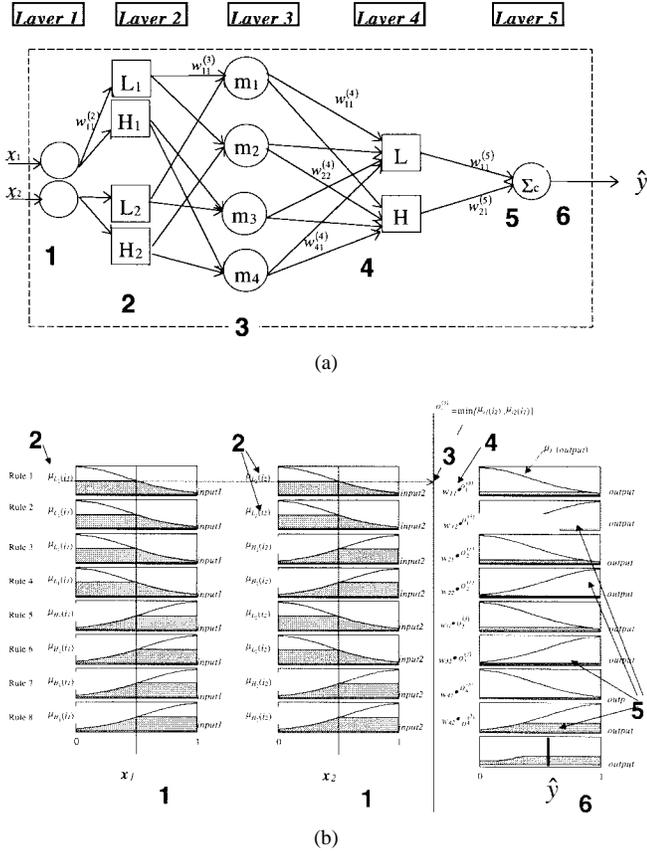


Fig. 3. The relationship of (a) the FZ/NN with (b) the FIS.

TABLE I
POINTWISE MATHEMATICAL INTERPRETATION OF THE FZ/NN SYSTEM

Point in Figure 3	Mathematical Interpretation
1	$o_j^{(1)} = u_j^{(1)}$
2	$o_j^{(2)} = e^{-\frac{(u_j^{(2)} - m_j)^2}{2\sigma_j^{(2)}}}$
3	$o_j^{(3)} = \min(u_1^{(3)}, u_2^{(3)}, \dots, u_n^{(3)})$
4	$u_i^{(4)} = w_{ij}^{(4)} o_j^{(3)}$
5	$o_j^{(5)} = \min\left(1, \sum_{i=1}^2 w_i^{(4)} u_i^{(4)}\right)$
6	$o_j^{(6)} = \frac{\sum_{i=1}^2 w_i^{(5)} u_i^{(5)}}{\sum_{i=1}^2 \sigma_i u_i^{(5)}}$

in input, output, membership function, membership function parameter, and rule parameter spaces, respectively.

IV. ENFORCING HEURISTIC CONSTRAINTS ON MEMBERSHIP FUNCTIONS

As discussed above, fuzzy/neural architectures (FZ/NN) are utilized to find, through training, a fuzzy inference system (FIS) that represents the underlying system, while also satisfying an error criterion. In most of the cases, using output error measures to validate extracted information is not sufficient, as extracted knowledge may not make heuristic sense even if the output error may meet the specified criteria.

TABLE II
FUZZY INFERENCE SYSTEM SPECIFICATIONS OF THE FZ/NN

Membership functions	Gaussian
Parameterization	Mean, variance
Fuzzy intersection	Minimum
Fuzzy inference operator	Correlation-Minimum (Mamdani's Operator)
Aggregation	Addition
Defuzzification	Center of area

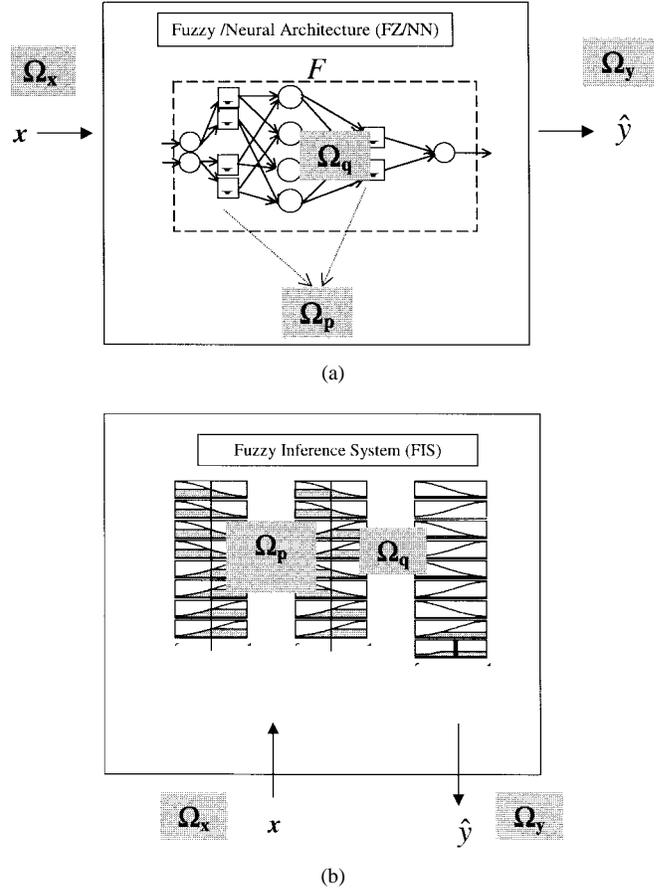


Fig. 4. The conceptual diagram of (a) the fuzzy/neural architecture and (b) the fuzzy inference system.

For any given problem, although information specific to the problem may not be available, information about the general properties of the solution may exist. For example, let us have three fuzzy sets on Ω_{x_1} , corresponding to low, medium, and high sets. For the membership that represents the “low” set μ_L , if $x_a \geq x_b$, heuristically it is expected that $\mu_L(x_a) \leq \mu_L(x_b)$, i.e., a lower value in the universe of discourse has a higher grade of membership to the “low” set. The training of the FZ/NN (without constraint enforcement) does not incorporate such *a priori* information and, therefore, the extracted knowledge in general may not conform to the *a priori* information. In this paper, a method to incorporate such information about the extracted knowledge into the training and rule extraction procedure for a fuzzy/neural system is proposed. This method is called the *heuristic constraints enforcement* method.

The aim of heuristic constraints enforcement is to achieve heuristically acceptable solutions for membership functions in

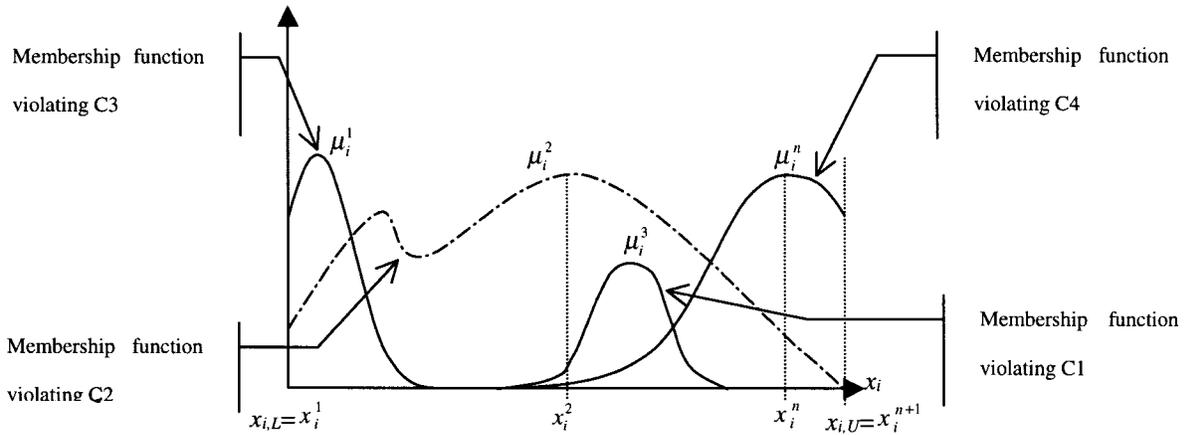


Fig. 5. Membership functions violating constraints C1–C4.

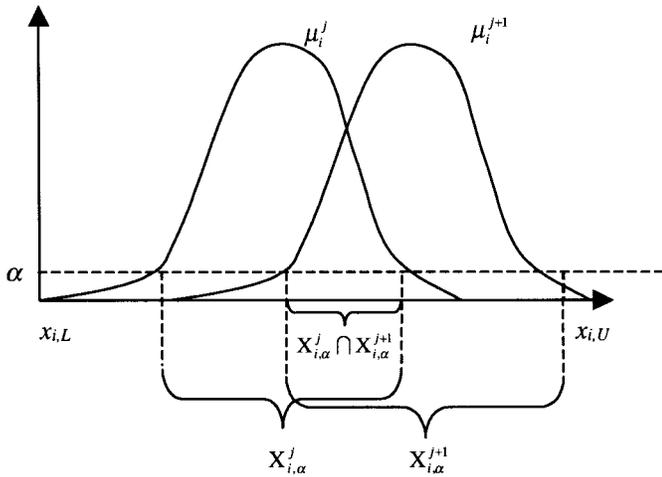


Fig. 6. The regions of interest for (17).

the input and output spaces, while meeting an error criterion defined on the output. The enforcement of constraints on membership functions, when integrated into the process of finding a rulebase, is expected to result in extraction of heuristically acceptable solutions for the membership functions representing fuzzy sets.

A. Heuristic Constraints on Membership Functions

Using heuristic constraints enforcement method, *a priori* information about the properties of the solution in terms of membership function properties is incorporated into the training of the FZ/NN. As described in [23], *a priori* information about the properties of a solution can be used to determine the acceptability of a solution. Analogously, we define constraints related to the fuzzy partitions of the input and output spaces, each of which reflect existing information that is used to evaluate the heuristic acceptability of the extracted information.

Using such *a priori* information, the following constraints are constructed. However, it is possible to add or remove constraints according to the requirements from the solution, which are determined by the nature of the specific problem

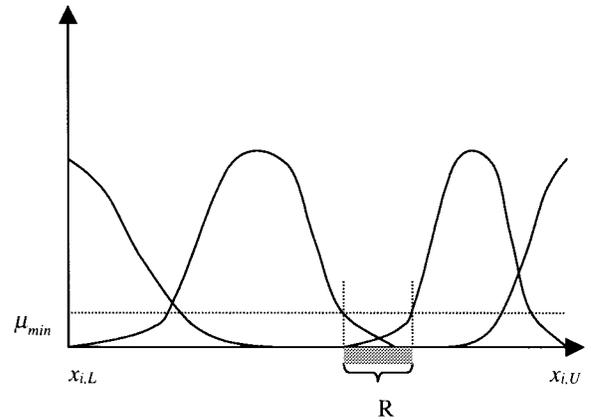


Fig. 7. A region $R \subset X_i$, where $x_i \in R$ violates (17).

at hand. The resulting constraint sets and the methods for enforcement of the constraints are given in Part II [24].

1) *Existence of Prototype Point Constraint:* At least one prototype point exists for each fuzzy set on X_i , which can be used to characterize each fuzzy set. In other words, each membership function attains a maximum value of one in the universe of discourse

$$\max_{x_i \in \Omega_{x_i}} (\mu_i^j(x_i)) = 1; \quad \forall j \in J. \quad (12)$$

2) *Convexity Constraint:* The membership functions must be convex, following Zadeh’s definition [25]; that is, the membership functions should be either nondecreasing, non-increasing, or have a nondecreasing portion followed by a nonincreasing portion

$$x_a \leq x_c \leq x_b \Rightarrow \min \{ \mu_i^j(x_a), \mu_i^j(x_b) \} \leq \mu_i^j(x_c); \quad \forall x_a, x_b, x_c \in \Omega_{x_i}; \quad \forall j \in J. \quad (13)$$

It is noted that the definition of convexity of membership functions is different from the conventional definition of convexity.

3) *Leftmost Membership Function Constraint:* The “leftmost” membership function, i.e., the membership function with the smallest x_i^j value as given in (9), is required to attain

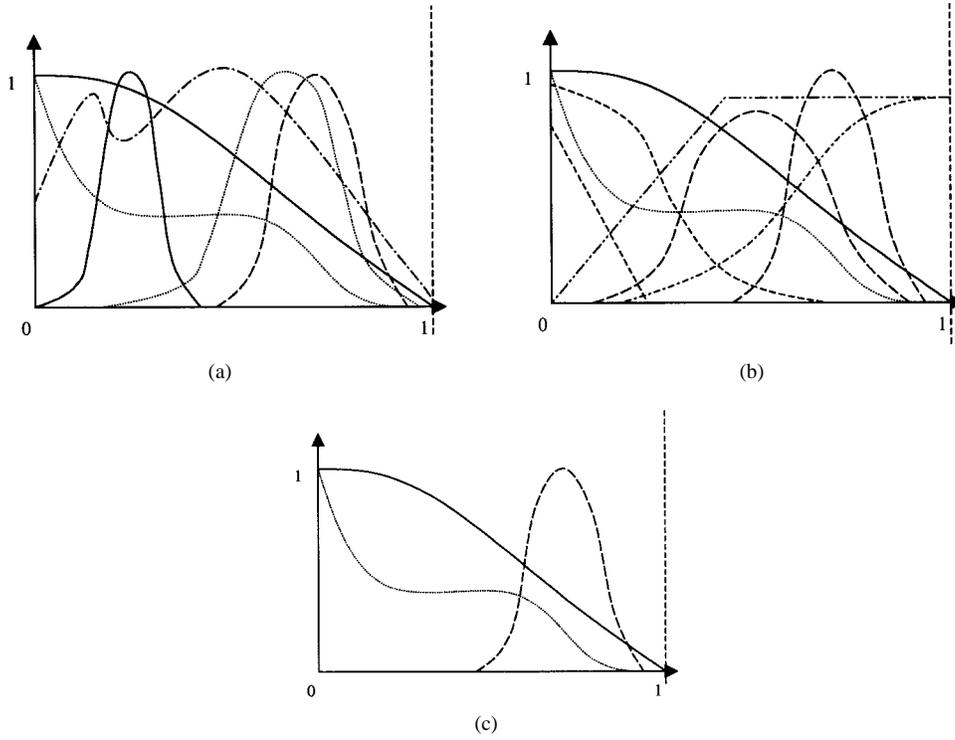


Fig. 8. Several elements of the constraint sets (a) C_1 , (b) C_2 , (c) $C_1 \cap C_2$.

its maximum at the smallest value of \mathbf{x}_i in $\Omega_{\mathbf{x}_i}$, which is $x_{i,L}$, i.e., maximum occurs at the smallest input value

$$x_i^k = x_{i,L}; \quad \forall x_i^k \ni (x_i^k < x_i^j) \quad \forall (j, k \in J, k \neq j). \quad (14)$$

4) *Rightmost Membership Function Constraint*: The “right-most” membership function, or the membership function with the largest x_i^j value given in (9), is required to attain its maximum at the largest value of \mathbf{x}_i in $\Omega_{\mathbf{x}_i}$, which is $x_{i,U}$

$$x_i^k = x_{i,U}; \quad \forall x_i^k \ni (x_i^k > x_i^j) \quad \forall (j, k \in J, k \neq j). \quad (15)$$

Several membership functions that do not agree with constraints (12), (13), (14), or (15) shown in Fig. 5.

5) *Overlap Constraint*: The overlap between the fuzzy sets should be in an interval $[L, U]$, $L \neq U$

$$0 < L \leq \frac{\text{diam}(X_{i,\alpha}^j \cap X_{i,\alpha}^{j\pm 1})}{\text{diam}(X_{i,\alpha}^j)} \leq U < 1$$

$$\forall j, \quad (j+1) \in J \quad \text{or} \quad j, (j-1) \in J \quad (16)$$

where j and $j \pm 1$ are indexes representing *adjacent* fuzzy sets, sorted according to the order of x_i^j given in (9). $X_{i,\alpha}^j$ is the α -cut set of X_i^j and “diam” is the diameter function [23] for the crisp α -cut sets. (16) is the overlap constraint for X_i^j . The regions of overlap are shown in Fig. 6. It is noted that the choice of L and U is important.

Very high values of U (or very small values of L) corresponding to large (or small) overlap between adjacent membership functions may be undesirable in system modeling and control applications [26]. Yet, for specific problems, the information about low (or high) overlap may be used to make inferences about necessity of architectural changes (e.g.,

adding or removing membership functions). Consequently, for problems where enforcing a lower and/or upper limit on the overlap is not advantageous, this constraint may be relaxed.

6) *Characterization Constraint*: Every point x_i in the universe of discourse X_i must have a grade of membership of at least $\mu_{\min} > 0$, in at least one of fuzzy sets X_i^j defined on X_i , so that every point in the universe can be characterized using (at least) one linguistic label attached to the fuzzy sets, i.e., every point in the universe of discourse can be considered to be a member of at least one fuzzy set with a minimum grade of membership of μ_{\min}

$$\min_{j \in J} \mu_i^j(x_i) = \mu_{\min}; \quad \forall x_i \in X_i. \quad (17)$$

Fig. 7 presents a region R in X_i where (17) is violated.

The list of constraints is subjective. According to the expectations from the solution, new constraints can be added, or some constraints can be deleted. When defining the constraints, it must be remembered that defining more constraints leads to incorporation of more *a priori* information into the problem.

B. Constraints on Total Solution in $\Omega_{\mathbf{y}}$

The magnitude of the error in the output space depends on the membership function parameters, as the output of the FZ/NN is a function of the parameters. The definition of the *total solution set* in the membership function parameter space, the elements of which provide a specified input–output mapping within a specified accuracy is given as follows.

Definition (Total Solution Set): Given a mapping (x, y) and the rule parameter vector \mathbf{q} , the *total solution set* on $\Omega_{\mathbf{p}}$ is defined as $\Theta_{q,\varepsilon} \subset \Omega_{\mathbf{p}}$, the elements of which satisfy the output

criterion set forth on \hat{y} in the output space Ω_y

$$\Theta_{q,\varepsilon} = \{\mathbf{p} \mid \|f(x, \mathbf{p}, \mathbf{q}) - y\| \leq \varepsilon; (x, y) \text{ in training set}\}. \quad (18)$$

The size of the total solution set depends on the choice of ε . The FZ/NN training strategy by itself does not guarantee convergence for arbitrary ε . Furthermore, a $\mathbf{p} \in \Theta_{q,\varepsilon}$ satisfying (12) is *not* guaranteed to conform to the heuristic knowledge in general. The motive for enforcing constraints is to extract information from FZ/NN that conforms to *a priori* information (e.g., heuristics about the problem) while satisfying (18).

C. Constraint Sets and Candidate Solutions

Each constraint defines a *constraint set* of elements satisfying the constraint, indicated by C_i , $1 \leq i \leq S$, where S is the number of constraints. Several candidate elements of C_1 , defined by constraint (12), are depicted in Fig. 8(a). Similarly, several elements of C_2 defined by (13) are depicted in Fig. 8(b). The elements of the intersection of these two sets are depicted in Fig. 8(c). As the membership functions must agree with both constraints (12) and (13), each membership function must be in both C_1 and C_2 . Consequently, it is concluded that any element that agrees with each constraint must actually be a member of the intersection of all the constraint sets, or, equivalently, $\mu \in \bigcap_{i=1}^S C_i$.

Definition (Candidate Solution Set): Following the definition of [20], we define a *candidate solution* in the membership function parameter space Ω_p as a $\mathbf{p} \in \Omega_p$, which conforms to the all heuristic knowledge incorporated in the form of constraints. The *candidate solution set* Γ is the set that contains all such elements in Ω_p or $\Gamma = \bigcap_{i=1}^S C_i$.

Ideally, the candidate solution set Γ is a subset of the total solution set $\Theta_{q,\varepsilon}$ so that any solution in Γ is also in the total solution set $\Theta_{q,\varepsilon}$

$$\Gamma \subset \Theta_{q,\varepsilon} \subset \Omega_p. \quad (19)$$

However, we cannot generally assume that (19) holds. Moreover, given any ε convergence is not guaranteed; even if convergence is achieved, the point of convergence is not guaranteed to be in $\Theta_{q,\varepsilon}$. Still, although $\mathbf{p} \in \Gamma$ may not be in $\Theta_{q,\varepsilon}$, every element $\mathbf{p} \in \Gamma$ is a candidate solution. Thus, if a choice of ε enables convergence in $\Theta_{q,\varepsilon}$, then the training with heuristic constraint enforcement is ensured to converge to a *candidate solution* in $\Theta_{q,\varepsilon}$. Because every element of the candidate solution set Γ complies with all constraints, every element of Γ is also an element of all the constraint sets C_i , $1 \leq i \leq S$. Consequently, the problem of finding a point $\mathbf{p} \in \Omega_p$ that satisfies all the constraints is actually equivalent to finding common point of the constraint sets C_i .

V. DISCUSSION

Part I of this paper presented the foundations of the proposed method of incorporating heuristic information in the form of constraints called heuristic constraint enforcement. The proposed method is not problem or architecture dependent; it can be applied to other realizations of a fuzzy inference

system including a conventional fuzzy inference system. Yet, the heuristic constraints are problem specific and they depend on the heuristic expectations from the solution. The collection of constraints given in this paper can be extended or condensed. Part II of this paper presents a method of finding candidate solutions defined in this paper and the technique of integrating the enforcement methodology with the training of the fuzzy/neural architecture together with applications.

REFERENCES

- [1] M. Y. Chow and P. V. Goode, "Hybrid fuzzy/neural system used to extract heuristic knowledge from a fault detection problem," in *IEEE Int. Conf. Fuzzy Syst.*, Orlando, FL, June 1994, vol. 3, pp. 1731–1736.
- [2] S. Altug, M. Y. Chow, and H. J. Trussell, "Fuzzy neural architectures for motor fault detection—Part I: Basic principles," *IEEE Trans. Indust. Electron.—Special Issue Fusion Neural Networks, Fuzzy Syst., Genetic Algorithms*, to be published.
- [3] ———, "Fuzzy neural architectures for motor fault detection—Part II: Comparative analysis for three-phase induction motor fault detection under varying load," *IEEE Trans. Indust. Electron.—Special Issue Fusion Neural Networks, Fuzzy Syst., Genetic Algorithms*, to be published.
- [4] S. Altug and M. Y. Chow, "Comparative analysis of fuzzy inference systems implemented on neural structures," in *Int. Conf. Neural Networks*, Houston, TX, June 1997, vol. 1, pp. 421–426.
- [5] I. Enbutsu, K. Baba, and N. Hara, "Fuzzy rule extraction from a multilayered neural network," in *Proc. IEEE Int. Joint Conf. Neural Networks*, Seattle, WA, July 1991, pp. 461–465.
- [6] L. M. Fu, "Rule generation from neural networks," *IEEE Trans. Syst., Man, Cybern.*, vol. 24, pp. 1114–1124, Aug. 1994.
- [7] H. Ishibuchi, R. Fujioka, and H. Tanaka, "Neural networks that learn from fuzzy If-Then rules," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 85–97, May 1993.
- [8] N. K. Kasabov, "Hybrid fuzzy connectionist rule-based systems and the role of fuzzy rules extraction," in *4th IEEE Int. Conf. Fuzzy Syst.*, Yokohama, Japan, Mar. 1995, vol. 1, pp. 49–56.
- [9] E. Khan and P. Venkatapuram, "Neufuz: Neural network based fuzzy logic design algorithms," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, San Francisco, CA, Mar. 1993, vol. 1.
- [10] C. Li and C.-H. Wu, "Automatically generated rules and membership functions for a neural fuzzy-based fault classifier," in *Midwest Symp. Circuits, Syst.*, Lafayette, LA, Aug. 1994, vol. 2, pp. 1377–1380.
- [11] C. T. Lin and C. S. G. Lee, "Neural-network-based fuzzy logic control and decision system," *IEEE Trans. Comput.*, vol. 40, pp. 1320–1336, Dec. 1991.
- [12] D. Nauck and R. Kruse, "A fuzzy neural network learning control rules and membership functions by fuzzy error propagation," in *Proc. IEEE Int. Conf. Neural Networks*, San Francisco, CA, Mar. 1993, vol. 2, pp. 1022–1027.
- [13] J. Nie, "Constructing fuzzy model by self-organizing counter propagation network," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, pp. 963–970, June 1995.
- [14] H. Nomura, I. Hayashi, and N. Wakanami, "A learning method of fuzzy inference rules by descent method," in *Proc. Int. Joint Conf. Neural Networks*, San Diego, CA, Mar. 1992.
- [15] C. W. Omlin and C. L. Giles, "Extraction of rules from discrete-time recurrent neural networks," *Neural Networks*, vol. 9, no. 1, pp. 41–52, 1996.
- [16] C. T. Sun, "Rule-base structure identification in an adaptive-network-based fuzzy inference system," *IEEE Trans. Fuzzy Syst.*, vol. 2, pp. 64–73, Feb. 1994.
- [17] L. X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Trans. Syst., Man, Cybern.*, vol. 22, pp. 665–685, Nov./Dec. 1992.
- [18] C.-T. Lin and C. S. G. Lee, *Neural Fuzzy Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [19] S. Abe and M. S. Lan, "Fuzzy rules extraction directly from numerical data for function approximation," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, pp. 119–129, Jan. 1995.
- [20] H. J. Trussell and M. R. Civanlar, "Feasible solution in signal restoration," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 201–212, Apr. 1984.
- [21] H. J. Trussell, "Convergence criteria for iterative restoration methods," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 129–136, Feb. 1983.

- [22] _____, "Application of set theoretic models to color systems," *Color Res. Applicat.*, vol. 16, no. 1, pp. 31–41, 1991.
- [23] P. L. Combettes, "The foundations of set theoretic estimation," *Proc. IEEE*, vol. 81, pp. 182–208, Feb. 1993 .
- [24] S. Altug, M. Y. Chow, and H. J. Trussell, "Heuristic constraints enforcement for training of and knowledge extraction from a fuzzy/neural architecture—Part II: Implementation and application," *IEEE Trans. Fuzzy Systems*, this issue, pp. 151–159.
- [25] L. A. Zadeh, "Fuzzy Sets," *Inform. Contr.*, vol. 8, pp. 338–353, 1965.
- [26] B. Kosko, *Neural Networks and Fuzzy Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1992.



Mo-Yuen Chow (S'81–M'87–SM'93) received the B.S. degree from the University of Wisconsin-Madison (electrical and computer engineering), and the M.Eng. and Ph.D. degrees from Cornell University, Ithaca, NY, in 1982, 1983, and 1987, respectively.

In 1987, he joined the faculty of North Carolina State University, Raleigh, NC, where he is presently an Associate Professor in the Department of Electrical and Computer Engineering. His core technologies are fault diagnosis and control, artificial neural networks and fuzzy logic. He has established the Advanced Diagnosis and Control Laboratory, as well as the FasPro Research Group. He has published one book, four book chapters, and over 80 journal and conference articles related to his research work.

Dr. Chow is an Ad Com member of the IEEE Industrial Electronics Society, an Ad Com member of the IEEE Neural Network Council, and Chairman of IEEE Neural Network Council Regional Interest Group Committee. He is also an Associate Editor for the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS.



Sinan Altug (S'96) was born in Ankara, Turkey, in 1971. He received the B.S. and M.S. degrees in electrical engineering from the Middle East Technical University, Turkey, in 1992 and 1994, respectively, the M.B.A. degree from Bilkent University, Turkey, in 1994, and the Ph.D. degree in electrical engineering from North Carolina State University, Raleigh, in 1998.

He worked as a Teaching and Research Assistant in North Carolina State University from 1995 through 1998. He is currently a European

Applications Manager at Champion Technologies, Inc.. His research interests include fusion of neural network and fuzzy logic technologies and motor fault detection and diagnosis.



H. Joel Trussell (S'75–M'76–SM'91–F'94) received the B.S. degree from Georgia Institute of Technology, Atlanta, in 1967, the M.S. degree from Florida State University, Tallahassee, in 1968, and the Ph.D. degree from the University of New Mexico, Albuquerque, in 1976.

He joined the Los Alamos Scientific Laboratory, Los Alamos, NM, in 1969 where he began working in the image and signal processing in 1971. From 1978 to 1979, he was a Visiting Professor at Heriot-Watt University, Edinburgh, Scotland, U.K., where he worked with both the university and with industry on image processing problems. In 1980, he joined the Electrical and Computer Engineering Department at North Carolina State University, Raleigh. During 1988–1989, he was a Visiting Scientist at the Eastman Kodak Company, Rochester, NY. From 1997 to 1998 he was a Visiting Scientist at Color Savvy Systems, Springboro, OH. His research has been in estimation theory and signal and image restoration.

He was a member and past chairman of the Image and Multidimensional Digital Signal Processing Committee of the Signal Processing Society of the IEEE. He was elected and served on the board of Governors of the Signal Processing Society from 1994 to 1997. He is the corecipient of the IEEE ASSP Society Senior Paper Award (1986, with M. R. Civanlar) and the IEEE Signal Processing Society Paper Award (1993, with P. L. Combettes). He is currently an associate editor for SIGNAL PROCESSING LETTERS and is a past associate editor for the IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING.