Unified Analysis and Design of ART/SOM Neural Networks and Fuzzy Inference Systems Based on Lattice Theory

Vassilis G. Kaburlasos

Technological Educational Institution of Kavala
Department of Industrial Informatics
65404 Kavala, Greece
vgkabs@teikav.edu.gr

Abstract. Fuzzy interval numbers (FINs, for short) is a unifying data representation analyzable in the context of lattice theory. This work shows how FINs improve the design of popular neural/fuzzy paradigms.

Keywords: Adaptive Resonance Theory (ART), Self-Organizing Map (SOM), Neural Networks, Fuzzy Inference System (FIS), Lattice Theory.

1 Introduction

Lattice theory has been proposed lately in Computational Intelligence (CI) with the potential to both unify and cross-fertilize [4,8]. An objective of this paper is to present recent advances based on fuzzy interval numbers, or FINs for short.

A FIN is a unifying data representation used for fuzzy numbers, intervals, real numbers, probability distribution functions, etc. Rigorous analysis of FINs, towards an improved design, can be pursued based on *lattice theory*. A FIN can be interpreted as an (information) *granule* [13]. In conclusion, FINs can be employed for improving a number of popular neural- and fuzzy- paradigms including (fuzzy) adaptive resonance theory (ART) [7], self-organizing maps (SOMs) [6], and fuzzy inference systems (FISs) [5]. Two novelties of this work include, first, an analysis of *interval type-2* (IT2) fuzzy sets [12] and, second, an extension of the *fuzzy lattice reasoning* (FLR) algorithm based on a similarity measure function in the space of FINs.

The layout of this paper is as follows. Section 2 summarizes the operation of popular neural/fuzzy paradigms. Section 3 outlines fuzzy interval number (FIN) mathematics. Section 4 presents unified extensions and improvements. Finally, section 5 concludes by summarizing the contribution of this work.

2 Fuzzy-ART, SOM, and FIS Operation

This section illustrates the operation of three popular Computational Intelligence paradigms, namely fuzzy-ART, SOM, and FISs.

F. Sandoval et al. (Eds.): IWANN 2007, LNCS 4507, pp. 80–93, 2007.
© Springer-Verlag Berlin Heidelberg 2007

2.1 Fuzzy Adaptive Resonance Theory (Fuzzy-ART)

The original fuzzy-ART neural network for clustering regards a two-layer architecture [1]. Layer F1 of fuzzy-ART fans out an input vector to the fully-interconnected neurons in layer F2. A layer F2 neuron filters an input vector $\mathbf{x}$ by computing vector $\mathbf{x} \wedge \mathbf{w}$, where $\mathbf{w}$ is the *code* (vector) stored on interlayer links. More specifically, an entry of vector $\mathbf{x} \wedge \mathbf{w}$ equals the minimum of the corresponding (positive real number) entries of vectors $\mathbf{x}$ and $\mathbf{w}$. A version of algorithm fuzzy-ART for training is briefly described in the following.

Algorithm fuzzy-ART (for training)

ART-1: Do while there are more inputs.
Apply the *complement coding* technique to represent input $[x_{i,1}, ..., x_{i,N}] \in [0,1]^N$ by $\mathbf{x}_i = [x_{i,1}, ..., x_{i,N}, 1-x_{i,1}, ..., 1-x_{i,N}] \in \mathsf{R}^{2N}$, $i = 1, ..., n$. Then, present $\mathbf{x}_i$ to the (initially) "set" neurons in layer F2.

ART-2: Each layer F2 neuron with code $\mathbf{w}_j \in \mathsf{R}^{2N}$ computes its *choice* (*Weber*) function $T_j = |\mathbf{x}_i \wedge \mathbf{w}_j|/(\alpha + |\mathbf{w}_j|)$.

ART-3: If there are no "set" neurons in layer F2 then memorize input $\mathbf{x}_i$.
Else, competition among the "set" neurons in layer F2: Winner is neuron J such that $T_J \doteq arg\max_j T_j$.

ART-4: *Similarity Test*: $(|\mathbf{x}_i \wedge \mathbf{w}_J|/|\mathbf{x}_i|) \geq \rho$, where $|\mathbf{x}_i \wedge \mathbf{w}_J|/|\mathbf{x}_i|$ is the *match* function and $\rho \in (0,1]$ is the user-defined *vigilance parameter*.

ART-5: If the *Similarity Test* fails then "reset" the winner neuron; goto step ART-3 to search for a new winner.
Else, replace the winner neuron code $\mathbf{w}_J$ by $\mathbf{x}_i \wedge \mathbf{w}_J$; goto step ART-1.

We remark that $|\mathbf{x}|$ above equals, by definition, the sum of vector $\mathbf{x}$ (positive) entries. Parameter "α", in the *choice* (*Weber*) function T_j, is a very small positive number. After training each neuron defines a cluster by a hyperbox.

The corresponding testing phase is carried out by winner-take-all competition based on the *choice* (*Weber*) function.

It turns out that fuzzy-ART operates by conditionally enlarging hyperboxes in the unit $N-$dimensional hypercube. An input is always a trivial hyperbox, i.e. a $N-$dimensional point. By attaching class labels to hyperboxes, a neural network for classification emerges, namely *fuzzy-ARTMAP* (neural network).

2.2 Self-Organizing Map (SOM)

Kohonen's self-organizing map (SOM) architecture for clustering [10] includes a 2-dimensional $L \times L$ *grid* (or, *map*) of *neurons* (or, *cells*). Each cell $C_{i,j}$ stores a vector $\mathbf{m}_{i,j} = [m_{i,j,1}, ..., m_{i,j,N}]^T \in \mathsf{R}^N$, $i = 1, ..., L$, $j = 1, ..., L$. Vectors $\mathbf{m}_{i,j}$ are called *code vectors* and they are initialized randomly. A version of algorithm SOM for training is briefly described next.

Algorithm SOM (for training)

SOM-1: Initialize randomly the neurons on the $L \times L$ grid.
Repeat the following steps a user-defined number N_{epochs} of epochs, $t = 1, ..., N_{epochs}$.
SOM-2: For each training datum $\mathbf{x}_k \in \mathsf{R}^N$, $k = 1, ..., n$ carry out the following computations.
SOM-3: Calculate the Euclidean distance $d(\mathbf{m}_{i,j}, \mathbf{x}_k)$, $i, j \in \{1, ..., L\}$.
SOM-4: Competition among the neurons on the $L \times L$ grid: Winner is neuron $(I, J) \doteq arg \min_{i,j \in \{1,\ldots,L\}} d_1(\mathbf{m}_{i,j}, \mathbf{x}_k)$.
SOM-5: *Assimilation Condition*: Vector $\mathbf{m}_{i,j}$ is in the neighborhood of vector $\mathbf{m}_{I,J}$ on the $L \times L$ grid.
SOM-6: If the *Assimilation Condition* is satisfied then compute a new value $\mathbf{m}'_{i,j}$:

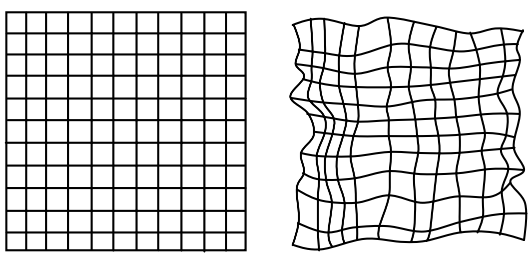
$$\mathbf{m}'_{i,j} = \mathbf{m}_{i,j} + a(t)(\mathbf{x}_k - \mathbf{m}_{i,j}) = [1 - a(t)]\mathbf{m}_{i,j} + a(t)\mathbf{x}_k, \tag{1}$$

where $a(t) \in (0, 1)$ is a decreasing function in time (t).

After training, each cell $C_{i,j}$ defines a cluster by code vector $\mathbf{m}_{i,j}$.

The corresponding testing phase is carried out by winner-take-all competition based on the Euclidean distance $d_1(.,.)$.

SOM operates by conditionally moving nodes on a 2-dimensional grid (Fig. 1). An input is always a $N-$dimensional point. By attaching class labels to nodes, a neural network for classification may emerge.



(a) (b)

Fig. 1. The SOM neural network for clustering operates by conditionally moving nodes on a 2-dimensional grid. (a) Initial node placement. (b) Node placement after training.

2.3 A Fuzzy Inference System (FIS)

A fuzzy inference system, or FIS for short, includes a knowledge base of fuzzy rules "if A_i then C_i", symbolically $A_i \to C_i$, $i = 1, \ldots, L$. Antecedent A_i is typically a conjunction of N fuzzy statements involving N fuzzy numbers, moreover consequent C_i may be either a fuzzy statement or an algebraic expression; the former is employed by a Mamdani type FIS based on expert knowledge [11], whereas the latter is employed by a Takagi-Sugeno-Kang (TSK) type FIS based on input-output measurements [14,15].

Based typically on fuzzy logic, a FIS input vector $\mathbf{x} \in \mathsf{R}^N$ activates in parallel rules in the knowledge-base by a *fuzzification* procedure; next, an *inference mechanism* produces the consequents of activated rules; the partial results are combined; finally, a real number vector is produced by a *de-fuzzification* procedure. Fig. 2 shows a Mamdani type FIS, involving triangular fuzzy membership functions in L fuzzy rules $R_1, \ldots, R_L$. The antecedent (IF part) of a rule is the conjunction of N fuzzy statements, whereas the consequent (THEN part) of a rule is a single fuzzy statement.

A FIS implements a function $f : \mathsf{R}^N \rightarrow \mathsf{K}$, where K is either discrete or continuous; e.g. the Mamdani FIS in Fig. 2 implements a function $f : \mathsf{R}^N \rightarrow \mathsf{R}$.

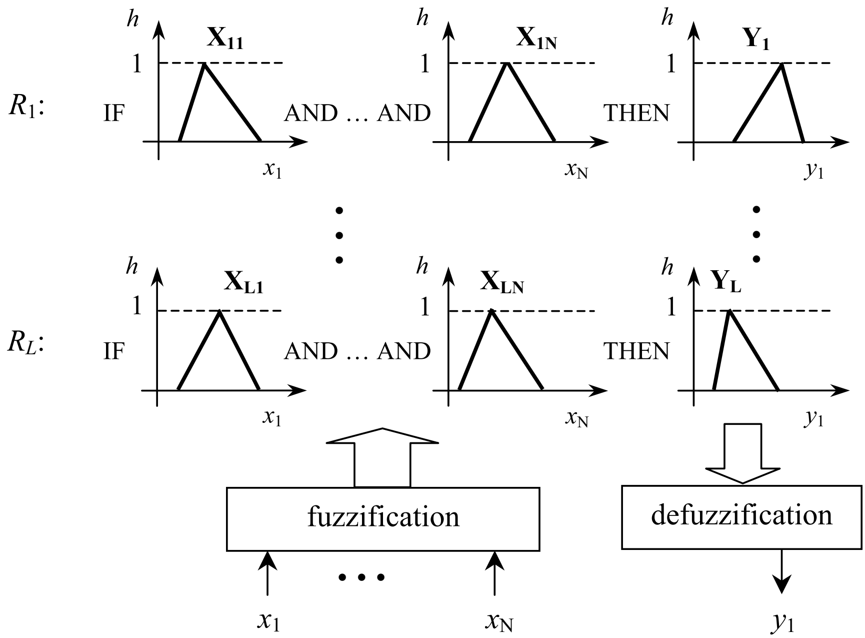


Fig. 2. A Mamdani type FIS with N inputs $x_1, \ldots, x_N$, one output y_1, and L fuzzy rules $R_1, \ldots, R_L$. This FIS, including both a fuzzification and a defuzzification procedure, implements a function $f : \mathsf{R}^N \rightarrow \mathsf{R}$.

Various FISs have been developed for inducing a function $f : \mathsf{R}^N \rightarrow \mathsf{R}^M$ from n pairs $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ of training data vectors. It turns out that the design of a FIS typically boils down to a parameter optimization problem, which seeks minimization of the LSE (error) function $\sqrt{\sum_{i=1}^{N} ||f(x_i) - y_i||}$. In particular, the design of a FIS concerns, first, an estimation of parameters which specify both the location and shape of fuzzy sets involved in the (fuzzy) rules of a FIS and, second, it may also concern the computation of the parameters of consequent algebraic equations in a TSK type FIS.

3 Fuzzy Interval Number (FIN) Mathematics

Some elementary mathematical lattice definitions are summarized next.

A *lattice* is a partially ordered set $(\mathsf{L}, \leq)$ any two of whose elements have both a *greatest lower bound*, denoted by $x \wedge y$, and a *least upper bound*, denoted by $x \vee y$. If $x \leq y$ (or, $y \leq x$) then two lattice elements x and y are called *comparable*; otherwise, x and y are called *incomparable*, symbolically $x||y$.

A useful function in a lattice $(\mathsf{L}, \leq)$ is a *positive valuation* function $v : \mathsf{L} \to \mathsf{R}$, which (by definition) satisfies (1) $v(x) + v(y) = v(x \wedge y) + v(x \vee y)$, and (2) $x < y \Rightarrow v(x) < v(y)$. Note that a positive valuation in a crisp lattice $(\mathsf{L}, \leq)$ implies a *metric* function $d : \mathsf{L} \times \mathsf{L} \to \mathsf{R}_0^+$ given by $d(x, y) = v(x \vee y) - v(x \wedge y)$.

Given (1) a product lattice $(\mathsf{L}, \leq) = (\mathsf{L}_1, \leq_1) \times \ldots \times (\mathsf{L}_N, \leq_N)$, and (2) a positive valuation $v : \mathsf{L}_i \to \mathsf{R}$ in each constituent lattice $(\mathsf{L}_i, \leq_i)$, $i = 1, ..., N$ then both a positive valuation $v : \mathsf{L} \to \mathsf{R}$ is given by $v(x_1, \ldots, x_N) = v_1(x_1) + \ldots + v_N(x_N)$, and countably infinite *Minkowski metrics* d_p are given in L by

$$d_p(\mathbf{x}, \mathbf{y}) = [d_1^p(x_1, y_1) + ... + d_N^p(x_N, y_N)]^{1/p}, \tag{2}$$

where $p = 1, 2, \ldots$ and $d_i(x_i, y_i) = v_i(x_i \vee y_i) - v_i(x_i \wedge y_i)$, $x_i, y_i \in \mathsf{L}_i$, $i = 1, \ldots, N$.

Of particular interest here is lattice $(\tau(\mathsf{L}), \leq)$, where $\tau(\mathsf{L})$ denotes the set of intervals[1] in L partially-ordered by set-inclusion. The *diagonal* of a lattice interval in $\tau(\mathsf{L})$ is defined as follows.

Definition 1. *Let $(\mathsf{L}, \leq)$ be a lattice. The* diagonal *of an interval $[a, b] \in \tau(\mathsf{L})$, $a, b \in \mathsf{L}$ with $a \leq b$, is defined as a nonnegative real function* $\text{diag}_p : \tau(\mathsf{L}) \to \mathsf{R}_0^+$ *given by $diag_p([a, b]) = d_p(a, b)$, $p = 1, 2, ...$*

In the following we focus on lattices stemming from the set R of real numbers. It turns out that $(\mathsf{R}, \leq)$ is a lattice including only comparable elements. Hence, lattice $(\mathsf{R}, \leq)$ is called *totally-ordered* or, equivalently, *chain*. In chain $(\mathsf{R}, \leq)$ any strictly increasing function $v_h : \mathsf{R} \to \mathsf{R}$ is a *positive valuation*, whereas any strictly decreasing function $\theta_h : \mathsf{R} \to \mathsf{R}$ is an *isomorphic* function[2] .

3.1 Generalized Intervals and Extensions

Generalized intervals are a basic instrument for FIN analysis, later.

[1] An *interval* $[a, b]$ is defined as the set $[a, b] \doteq \{x : a \leq x \leq b\}$.

[2] Given two lattices $(\mathsf{L}_1, \leq)$ and $(\mathsf{L}_2, \leq)$ a function $\psi : \mathsf{L}_1 \to \mathsf{L}_2$ is called *isomorphic* if both "$x \leq y$ in $\mathsf{L}_1 \Leftrightarrow \psi(x) \leq \psi(y)$ in L_2" and "ψ is onto L_2"

Definition 2. *(a) A* positive generalized interval *of height h is a map $\mu^h_{a,b}$: $\mathsf{R} \to \{0,h\}$ given by* $\mu^h_{a,b}(x) = \begin{cases} h, & a \le x \le b \\ 0, & otherwise \end{cases}$, *where $h \in (0,1]$. (b) A* negative generalized interval *of height h is a map $\mu^h_{a,b} : \mathsf{R} \to \{0,-h\}$ given by* $\mu^h_{a,b}(x) = \begin{cases} -h, & a \ge x \ge b \\ 0, & otherwise \end{cases}$, *where $a > b$ and $h \in (0,1]$.*

We remark that a generalized interval is a "box" function, either positive or negative. In the interest of simplicity a generalized interval will be denoted as $[a,b]^h$, where $a \le b$ ($a > b$) for a positive (negative) generalized interval.

The set of positive (negative) generalized intervals of height h is denoted by $\mathsf{M}^h_+(\mathsf{M}^h_-)$. The set of generalized intervals of height h is denoted by M^h, i.e. $\mathsf{M}^h = \mathsf{M}^h_- \cup \mathsf{M}^h_+$. It turns out that the set M^h of generalized intervals is *partially ordered*; more specifically, M^h is a *mathematical lattice* [4] with lattice *meet* and *join* given, respectively, by $[a,b]^h \wedge [c,d]^h = [a \vee c, b \wedge d]^h$ and $[a,b]^h \vee [c,d]^h = [a \wedge c, b \vee d]^h$. Moreover, the corresponding lattice order relation $[a,b]^h \le [c,d]^h$ in M^h is equivalent to "$c \le a$".AND."$b \le d$" (Fig. 3).

Given both a strictly increasing function $v_h : \mathsf{R} \to \mathsf{R}$ and a strictly decreasing function $\theta_h : \mathsf{R} \to \mathsf{R}$, a positive valuation in lattice $(\mathsf{M}^h, \le)$ is given by $v_{\mathsf{M}^h}([a,b]^h) = v_h(\theta_h(a)) + v_h(b)$. Hence, a metric in lattice $(\mathsf{M}^h, \le)$ is given by $d_{\mathsf{M}^h}([a,b]^h, [c,d]^h) = [v_h(\theta_h(a \wedge c)) - v_h(\theta_h(a \vee c))] + [v_h(b \vee d) - v_h(b \wedge d)]$. For example, choosing both $\theta_h(x) = -x$ and v_h such that $v_h(x) = -v_h(-x)$ it follows positive valuation function $v_{\mathsf{M}^h}([a,b]^h) = v_h(b) - v_h(a)$; furthermore, it follows metric $d_{\mathsf{M}^h}([a,b]^h, [c,d]^h) = [v_h(a \vee c) - v_h(a \wedge c)] + [v_h(b \vee d) - v_h(b \wedge d)]$.

The space M^h of generalized intervals is a real *linear space* with

- *addition* defined as $[a,b]^h + [c,d]^h = [a+c, b+d]^h$.
- *multiplication* (by $k \in \mathsf{R}$) defined as $k[a,b]^h = [ka, kb]^h$.

A subset C of a linear space is called *cone* if for all $x \in C$ and a positive real number $\lambda > 0$ we have $\lambda x \in C$. It turns out that both M^h_+ and M^h_- are cones.

Let $(\mathsf{M}^h, \le)^\partial = (\mathsf{M}^h, \le^\partial)$ denote the *dual* (lattice)[3] of lattice $(\mathsf{M}^h, \le)$. Then, $(\mathbb{M}^h, \le) = (\mathsf{M}^h \times \mathsf{M}^h, \le^\partial \times \le)$ is a lattice. In the following we introduce a positive valuation function in lattice $(\mathbb{M}^h, \le)$.

Proposition 1. *Let function $v_{\mathsf{M}^h} : \mathsf{M}^h \to \mathsf{R}$ be a positive valuation function in a lattice $(\mathsf{M}^h, \le)$. Then, function $v_{\mathbb{M}^h} : \mathsf{M}^h \times \mathsf{M}^h \to \mathsf{R}$ given by $v_{\mathbb{M}^h}([\kappa_h, \lambda_h]) = v_{\mathsf{M}^h}(\lambda_h) - v_{\mathsf{M}^h}(\kappa_h)$ is a positive valuation in lattice $(\mathbb{M}^h, \le)$.*

The proof of proposition 1 will be shown elsewhere for lack of space.

It follows a metric $d_{\mathbb{M}^h} : \mathbb{M}^h \times \mathbb{M}^h \to \mathsf{R}^+_0$ given by $d_{\mathbb{M}^h}([A,B]^h, [C,D]^h) = [v_{\mathsf{M}^h}(A \vee C) - v_{\mathsf{M}^h}(A \wedge C)] + [v_{\mathsf{M}^h}(B \vee D) - v_{\mathsf{M}^h}(B \wedge D)]$.

[3] The *dual* (denoted by $\ge$) of an order relation $\le$ is, by definition, the inverse of $\le$.

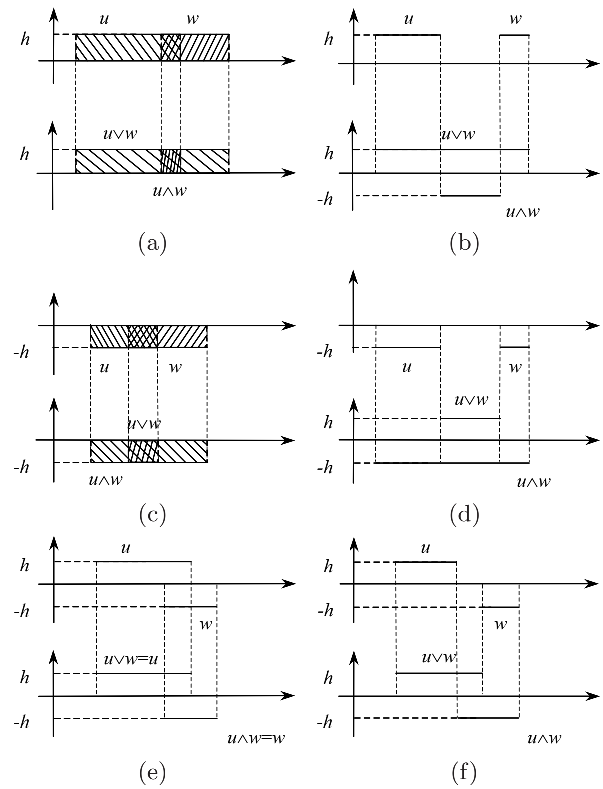


Fig. 3. Lattice- *join* $(u \vee w)$ and *meet* $(u \wedge w)$ for all different pairs of generalized intervals of height h. Different fill-in patterns are used for partially overlapped generalized intervals. (a) "Intersecting" positive generalized intervals. (b) "Nonintersecting" positive generalized intervals. (c) "Intersecting" negative generalized intervals. (d) "Nonintersecting" negative generalized intervals. (e) "Intersecting" positive and negative generalized intervals. (f) "Nonintersecting" positive and negative generalized intervals.

3.2 Fuzzy Interval Numbers (FINs)

Consider the following definition.

Definition 3. *A* Fuzzy Interval Number*, or* FIN *for short, is a function* $F : (0,1] \rightarrow \mathsf{M}$ *such that (1)* $F(h) \in \mathsf{M}^h$*, (2) either* $F(h) \in \mathsf{M}_+^h$ *(positive FIN), or* $F(h) \in \mathsf{M}_-^h$ *(negative FIN) for all* $h \in (0,1]$*, and (3)* $h_1 \leq h_2 \Rightarrow \{x : F(h_1) \neq 0\} \supseteqq \{x : F(h_2) \neq 0\}$.

A FIN F can be written as the set union of generalized intervals; in particular, $F = \bigcup_{h \in (0,1]} \{[a(h), b(h)]^h\}$, where both interval-ends $a(h)$ and $b(h)$ are functions of $h \in (0,1]$. The set of FINs is denoted by F. More specifically, the set of positive (negative) FINs is denoted by F_+ (F_-). Fig. 4 shows a positive FIN F_p.

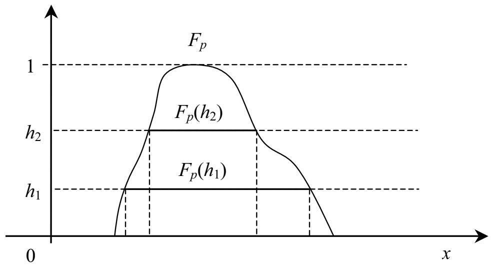


Fig. 4. A positive FIN $F_p = \bigcup_{h\in(0,1]} \{F_p(h)\}$ is the set-union of positive generalized intervals $F_p(h)$, $h \in (0,1]$ such that $h_1 \leq h_2 \Rightarrow \{x : F(h_1) \neq 0\} \supseteqq \{x : F(h_2) \neq 0\}$.

We define an *interval-FIN* as $F = \bigcup_{h\in(0,1]} \{[a(h), b(h)]^h\}$, where both $a(h)$ and $b(h)$ are constant, i.e. $a(h) = a$ and $b(h) = b$. In particular, for $a = b$ an interval-FIN is called *trivial-FIN*. In the aforementioned sense F_+ includes both (fuzzy) numbers and intervals.

We remark that a FIN is a mathematical object, which can be interpreted either as a possibility distribution (i.e. a fuzzy number) or as a probability distribution. In any case, a FIN can be interpreted as an (information) *granule*. Note that *granular computing* is considered an emerging computational paradigm [13].

An ordering relation has been introduced in F as follows: $F_1 \leq F_2 \Leftrightarrow F_1(h) \leq F_2(h), \forall h \in (0,1]$. It turns out that F is a mathematical lattice. The following proposition introduces a metric in F.

Proposition 2. *Consider metrics* $d_{\mathsf{M}^h} : \mathsf{M}^h \times \mathsf{M}^h \to \mathsf{R}_0^+$ *in lattices* $(\mathsf{M}^h, \leq)$, $h \in (0,1]$. *Let* $F_1, F_2 \in (\mathsf{F}, \leq)$. *A metric function* $d_{\mathsf{F}} : \mathsf{F} \times \mathsf{F} \to \mathsf{R}_0^+$ *is given by*

$$d_{\mathsf{F}}(F_1, F_2) = \int_0^1 d_{\mathsf{M}^h}(F_1(h), F_2(h))dh \tag{3}$$

Addition and multiplication are extended from M^h to F as follows.

- The *product* kF_1, where $k \in \mathsf{R}$ and $F_1 \in \mathsf{F}$, is defined as $F_p : F_p(h) = kF_1(h)$, $h \in (0,1]$.
- The *sum* $F_1 + F_2$, where $F_1, F_2 \in \mathsf{F}$ is defined as $F_s : F_s(h) = (F_1 + F_2)(h) = F_1(h) + F_2(h)$, $h \in (0,1]$.

We remark that the product kF_1 is always a FIN. It turns out that both F_+ and F_- are cones. When both F_1 and F_2 are in cone F_+ (F_-) then the sum $F_1 + F_2$ is in cone F_+ (F_-). However, if $F_1 \in \mathsf{F}_+$ and $F_2 \in \mathsf{F}_-$ then $F_1 + F_2$ might not be a FIN. Our interest here is in the *metric lattice cone* F_+ of positive FINs.

3.3 Interval Type-2 FINs

Generalized type-2 fuzzy sets, or simply *type-2 fuzzy sets*, are an extension of type-1 (regular) fuzzy sets such that the membership grade of a type-2 fuzzy set is a type-1 fuzzy set [16]. There is a growing interest in type-2 fuzzy systems [3].

Type-2 literature has predominantly become concerned with interval type-2 (IT2) fuzzy sets, that is a subset of type-2 fuzzy sets such that the membership grade is an interval in order to alleviate a number of computational problems.

Fig. 5 shows two convex fuzzy sets, i.e. $F_u(x)$ and $f_l(x)$, such that $f_l(x) \leq F_u(x)$, $\forall x \in \mathsf{R}$. We point out that intervals $[f_l(x), F_u(x)]$, $x \in \mathsf{R}$ can be used for representing a IT2 fuzzy set with *lower* and *upper* membership functions $f_l(x)$ and $F_u(x)$, respectively. In the aforementioned sense, a fuzzy set, of either type-1 or type-2, is described "vertically".

IT2 fuzzy sets are an interpretation of IT2 FINs introduced next.

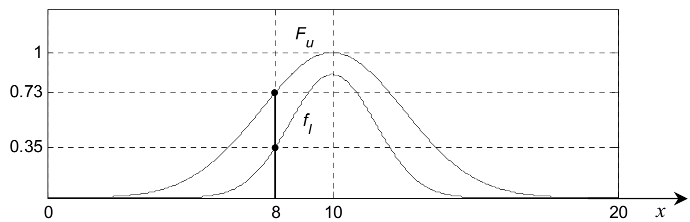


Fig. 5. Both *upper*- and *lower*- FIN membership functions are represented vertically. The corresponding IT2 FIN is described vertically by intervals $[f_l(x), F_u(x)]$, $x \in \mathsf{R}$.

Definition 4. *A* IT2 FIN *is an interval* $[A, B]$ *of FINs such that* $A \leq B$*.*

Fig. 6 shows two (convex) FINs interpreted as fuzzy sets.

The difference is that a FIN is described "horizontally" by the α-cuts of a fuzzy set defined (the α-cuts) between dots on a horizontal line, e.g. the line

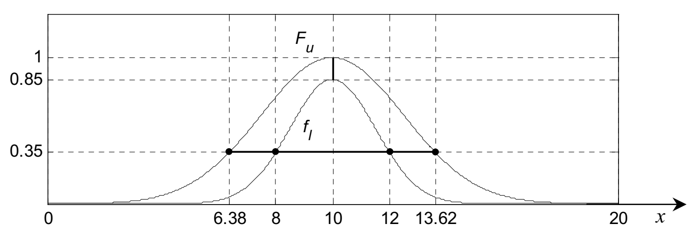


Fig. 6. Both *upper*- and *lower*- FIN membership functions are represented horizontally by lattice-ordered *generalized intervals* $f_l(h), F_u(h)$, $h \in (0, 1]$. The corresponding IT2 FIN is described horizontally by intervals $[f_l(h), F_u(h)]$, $h \in (0, 1]$. Note the extension inserted to the *lower* FIN membership function, at $x = 10$, in order to normalize it.

through membership grade 0.35 in Fig. 6. Note that an extension was inserted to the *lower* FIN membership function, at $x = 10$, in order to normalize it.

Based on the previous analysis it can be shown that the set $\mathbb{F}$, of IT2 FINs, is lattice-ordered. A IT2 FIN can be interpreted either as a IT2 fuzzy set or as an interval of probability distribution functions (PDFs). The following proposition introduces a metric in $\mathbb{F}$.

Proposition 3. *Consider metrics* $d_{\mathbb{M}^h} : \mathbb{M}^h \times \mathbb{M}^h \rightarrow \mathsf{R}_0^+$ *in lattices* $(\mathbb{M}^h, \leq)$, $h \in (0,1]$. *Let* $\mathcal{F}_1, \mathcal{F}_2 \in (\mathbb{F}, \leq)$. *A metric function* $d_{\mathbb{F}} : \mathbb{F} \times \mathbb{F} \rightarrow R_0^+$ *is given by*

$$d_{\mathbb{F}}(\mathcal{F}_1, \mathcal{F}_2) = \int_0^1 d_{\mathbb{M}^h}(\mathcal{F}_1(h), \mathcal{F}_2(h))dh \tag{4}$$

4 Unified Extensions and Improvements

Based on a Minkowski metric d_p of Eq. (2) above, this section delineates extensions of ART/SOM neural networks as well as of fuzzy inference systems (FISs) in the *metric lattice cone* F_+ of positive FINs.

4.1 Fuzzy Lattice Reasoning (FLR)

Algorithm fuzzy lattice reasoning (FLR) was described lately as a lattice data domain extension of fuzzy-ARTMAP based on a lattice *inclusion measure* function [9]. Note also that, lately, were presented versions of the FLR algorithm based on similarity measures instead of an inclusion measure function [2]. A likewise extension is presented in this section based on a similarity measure (function). A rigorous definition of the latter function is introduced next.

Definition 5. *A* similarity measure *in a set* S *is a function* $\tau : S \times S \rightarrow (0,1]$, *which satisfies the following conditions.*

(S1) $\tau(a,b) = 1 \Leftrightarrow a = b$.

(S2) $\tau(a,b) = \tau(b,a)$.

(S3) $\frac{1}{\tau(a,b)} + \frac{1}{\tau(x,x)} \leq \frac{1}{\tau(a,x)} + \frac{1}{\tau(x,b)}$.

We remark that condition $S1$ requires that two set S elements a and b are "most similar" to each other if and only if a and b coincide. Condition $S2$ requires that an element a is so much similar to an element b as b is to a (*Commutativity*). Condition $S3$ requires that if two elements a and b are "little" similar to each other, i.e. $0 < \tau(a,b) << 1$ and, moreover, an element x is "very" similar to one of a or b then x has to be "little" similar to the other one of a and b. A similarity measure function can be defined based on a metric as shown next.

Proposition 4. *If function* $d : S \times S \rightarrow R_0^+$ *is a metric then function* τ_d : $S \times S \rightarrow (0,1]$ *given by* $\tau_d(a,b) = 1/(1 + d(a,b))$ *is a similarity measure.*

Algorithm FLR for training

FLR-0: A set $RB = \{(u_1, C_1), ..., (u_L, C_L)\}$ is given, where $u_i \in \mathsf{F}_+^N$ and $C_i \in \mathsf{C}$, $i = 1, ..., L$ is a class label in the finite set C.

FLR-1: Present the next input pair $(x_i, c_i) \in \mathsf{F}_+^N \times \mathsf{C}, i = 1, ..., n$ to the initially "set" RB.

FLR-2: If no more pairs are "set" in RB then store input pair (x_i, c_i) in RB; $L \leftarrow L + 1$; goto step FLR-1.

Else, compute the similarity measure $\tau(x_i, u_l)$, $l \in \{1, ..., L\}$ regarding input $x_i \in \mathsf{F}_+^N$ to all "set" elements $u_i \in \mathsf{F}_+^N, i = 1, ..., L$ in RB.

FLR-3: Competition among the "set" pairs in the RB: Winner is pair (u_J, C_J) such that $J \doteq arg \max_{l \in \{1,...,L\}} \tau(x_i, u_l)$. In case of multiple winners, choose the one with the smallest diagonal $diag_1(.)$ size.

FLR-4: *Assimilation Condition*: Both (1) $diag_1(x_i \vee u_J)$ is less than a user-defined threshold size D_{crit}, and (2) $c_i = C_J$.

FLR-5: If the *Assimilation Condition* is not satisfied then "reset" the winner pair (u_J, C_J); goto step FLR-2.

Else, replace the winner hyperbox u_J by the join-interval $x_i \vee u_J$; goto step FLR-1.

We remark that the join-interval $x_i \vee u_J$ of two positive FINs $u_i, u_J \in \mathsf{F}_+$ is computed as $(x_i \vee u_J)(h) \doteq x_i(h) \vee u_J(h)$, $h \in (0, 1]$.

The corresponding testing phase is carried out by winner-take-all competition based on the similarity measure function $\tau(.,.)$.

There are inherent similarities as well as substantial differences between fuzzy-ARTMAP and the FLR. In particular, both fuzzy-ARTMAP and FLR carry out learning rapidly in a single pass through the training data by computing hyperboxes in their corresponding data domains.

Advantages of FLR over fuzzy-ARTMAP include (1) *granularity*, and (2) *flexibility* as summarized next. (1) The FLR handles positive FINs, including both (fuzzy) numbers and intervals in the Euclidean space R^N; whereas fuzzy-ARTMAP deals solely with intervals in the unit-hypercube. (2) It is possible to optimize FLR's behavior by tuning an underlying positive valuation function v as well as an isomorphic function θ; whereas fuzzy-ARTMAP implicitly uses, quite restrictively, only $v(x) = x$ and $\theta(x) = 1 - x$ in a data dimension.

Fuzzy-ARTMAP's *proliferation problem*, that is the proliferation of hyperboxes/clusters, is inherited to the FLR. However, the FLR is equipped with a metric function tool, hence it is possible to reduce "in principle" the number of its clusters in F_+^N.

Another drawback of FLR, also inherited from fuzzy-ARTMAP, is that the learned clusters (in particular their total number, size, and location) depend on the order of presenting the training data. A potential solution is to employ an ensemble of FLR classifiers in order to boost performance stably [4].

4.2 Incremental Granular SOM (grSOM)

A granular SOM (grSOM) algorithm for learning (training) is presented next as a straightforward extension of SOM in *metric lattice cone* F_+ of positive FINs.

A grSOM algorithm for training

GR-0: Define the size L of a $L \times L$ grid of neurons. Each neuron can store both a N-dimensional FIN $W_{i,j} \in \mathsf{F}_+^N$, $i, j \in 1, ..., L$ and a class label $C_{i,j} \in \mathsf{C}$, where C is a finite set. Initially all neurons are *uncommitted*.

GR-1: Memorize the first training data pair $(x_1, C_1) \in \mathsf{F}_+^N \times \mathsf{C}$ by committing, randomly, a *neuron* in the $L \times L$ grid.
Repeat the following steps a user-defined number N_{epochs} of epochs, $p = 1, ..., N_{epochs}$.

GR-2: For each training datum $(x_k, C_k) \in \mathsf{F}_+^N \times \mathsf{C}$, $k = 1, ..., n$ "reset" all $L \times L$ grid neurons. Then carry out the following computations.

GR-3: Calculate the Minkowski metric $d_1(x_k, W_{i,j})$ between x_k and *committed* neurons $W_{i,j}$ $i, j \in \{1, ..., L\}$.

GR-4: Competition among the "set" (and, *committed*) neurons in the $L \times L$ grid: Winner is neuron (I, J) whose weight $W_{I,J}$ is the nearest to x_k, i.e. $(I, J) \doteq arg \min_{i,j \in \{1,...,L\}} d_1(x_k, W_{i,j})$.

GR-5: *Assimilation Condition*: Both (1) Vector $W_{i,j}$ is in the neighborhood of vector $W_{I,J}$ on the $L \times L$ grid, and (2) $C_{I,J} = C_k$.

GR-6: If the *Assimilation Condition* is satisfied then compute a new value $W'_{i,j}$ as follows:
$$W'_{i,j} \doteq \left[1 - \frac{h(k)}{1+d_K(W_{I,J}, W_{i,j})}\right] W_{i,j} + \frac{h(k)}{1+d_K(W_{I,J}, W_{i,j})} x_k.$$
Else, "reset" the winner (I, J); goto GR-4.

GR-7: If all the $L \times L$ neurons are "reset" then commit an *uncommitted* neuron from the grid to memorize the current training datum (x_k, C_k).
If there are no more *uncommitted* neurons then increase L by one.

We remark that function $h(k)$, in the training phase step GR-6, reduces smoothly from 1 down to 0 with the epoch number k. The above algorithm is called, in particular, *incremental-grSOM*.

The corresponding testing phase is carried out by winner-take-all competition based on the Minkowski metric $d_1(.,.)$.

A fundamental improvement of the *incremental-grSOM* over SOM is the sound capacity of the *incremental-grSOM* to rigorously deal with granular data including both fuzzy numbers and intervals represented by FINs.

4.3 Novel FIS Analysis and Design

In contrast to alternative, "parametric" function estimation methods including statistical regressors, ARMA models, and multilayer perceptrons a FIS induces rules from the training data. Moreover, it is widely recognized that a FIS can produce better results than alternative function approximation methods and, usually, a fuzzy logic explanation is sought. A set-theoretic explanation has been proposed lately [5] by seeking an answer to the following question: How many fuzzy numbers are there? Or, in other words, *what is the cardinality (card(*$\mathsf{F_n}$*)) of the set* $\mathsf{F_n}$ of fuzzy numbers? It follows a non-obvious mathematical result.

Proposition 5. *It holds card*$(\mathsf{F_n}) = \aleph_1$, *where* $\aleph_1$ *is the cardinality of the set* R *of real numbers.*

In other words, there are as many fuzzy numbers as there are real numbers. Proposition 5 leads to novel perspectives regarding the capacity of FIS for function approximation as explained in the following.

In the first place it is interesting to calculate the cardinality of the set $\mathcal{F}$ of all functions $f : \mathsf{R}^N \rightarrow \mathsf{R}^M$. Using *cardinal arithmetic* it follows that $card(\mathcal{F}) = \aleph_1^{\aleph_1} = (2^{\aleph_0})^{\aleph_1} = 2^{\aleph_1} = \aleph_2 > \aleph_1$. Unfortunately a general function f_0 in $\mathcal{F}$ is practically useless because it lacks a capacity for generalization. More specifically, knowledge of a function f_0 values $f_0(x_1), ...f_0(x_n)$ at a number of points $x_1, ..., x_n$ cannot imply the value of function f_0 at a different point $x_{n+1} \neq x_i$, $i = 1, \dots, n$.

Consider now a parametric family of models, e.g. polynomials, ARMA models, statistical regressors, multilayer perceptrons, etc. Any of the aforementioned families is characterized by a capacity for generalization. Moreover, due to the finite number p of parameters involved, the cardinality of any of the aforementioned families of models equals $\aleph_1^p = (2^{\aleph_0})^p = 2^{\aleph_o} = \aleph_1$.

It might be thought that $\aleph_1$ is an adequately large number of models to choose a "good" model from. Unfortunately the latter is not the case. Consider, for instance, the family of polynomials which includes $\aleph_1$ models. It is well known that a polynomial may not approximate usefully a set $(x_1, y_1), ..., (x_n, y_n)$ of training data points due to "overfitting"; hence a polynomial may not be useful for generalization. It turns out that the family of FISs combines cardinality $\aleph_2$ with a capacity for (local) generalization as explained next.

It has been shown in proposition 5 that the cardinality of the set $\mathsf{F_n}$ of fuzzy numbers equals $card(\mathsf{F_n}) = \aleph_1$. A Mamdani type FIS can be regarded as a function $m : \mathsf{F_n}^N \rightarrow \mathsf{F_n}^M$. Using standard cardinal arithmetic it follows that the cardinality of the set M of Mamdani type FIS equals $card(\mathsf{M}) = \aleph_1^{\aleph_1} = \aleph_2 > \aleph_1$. In conclusion, Mamdani- type FIS can implement, in principle, $\aleph_2$ functions. The same is true of Sugeno-type FIS.

It was explained that a general function $f : \mathsf{R}^N \rightarrow \mathsf{R}^M$ lacks a capacity for generalization. Fortunately this is not the case for a FIS of Mamdani-, or Sugeno-type due to the non-trivial (interval) support of the fuzzy numbers involved in a FIS' fuzzy rule base. More specifically an input vector $x = (x_1, \dots, x_n)$, within a fuzzy rule's interval of support, activates the corresponding rule; there follows a FIS' capacity for (local) generalization. In conclusion the family of FIS models combines "in principle" a cardinality $\aleph_2$ with a capacity for generalization in function $f : \mathsf{R}^N \rightarrow \mathsf{R}^M$ approximation problems.

Note that, lately, FINs were used for novel FIS analysis and design based on metric topology techniques [5]. In addition, extensive statistical "hypothesis testing" has demonstrated that *genetically* optimized positive valuation functions can result in substantial improvement in applications using FINs [6].

5 Conclusion

This chapter was meant as a reference towards proliferating the employment of FINs in neural/fuzzy applications. In addition to a capacity to rigorously deal with granular inputs, an important advantage of the proposed techniques is the introduction of tunable nonlinearities based on positive valuation functions.

There is ample experimental evidence suggesting that FIN extensions of ART, SOM, and FIS can comparatively improve performance in classification and regression applications [4,5,6,7,9]. Furthermore, note that IT2 FIN extensions of both ART and SOM neural networks are straightforward in the context of this work.

References

1. Carpenter, G.A., Grossberg, S., Rosen, D.B.: Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. Neural Networks 4(6), 759–771 (1991)
2. Cripps, A., Nguyen, N.: Fuzzy lattice reasoning (FLR) classification using similarity measures. In: Kaburlasos, V.G., Ritter, G.X. (eds.) Computational Intelligence Based on Lattice Theory, Springer, Heidelberg (2007)
3. John, R., Coupland, S.: Extensions to type-1 fuzzy logic: Type-2 fuzzy logic and uncertainty. In: Yen, G. Y., Fogel, D. B. (eds.) Computational Intelligence: Principles and Practice pp. 89-101. IEEE Computational Intelligence Society (2006)
4. Kaburlasos, V.G.: Towards a Unified Modeling and Knowledge Representation Based on Lattice Theory — Computational Intelligence and Soft Computing Applications. Studies in Computational Intelligence, vol. 27. Springer, Heidelberg (2006)
5. Kaburlasos, V.G., Kehagias, A.: Novel fuzzy inference system (FIS) analysis and design based on lattice theory. IEEE Trans. Fuzzy Systems 15(2), 243–260 (2007)
6. Kaburlasos, V.G., Papadakis, S.E.: Granular self-organizing map (grSOM) for structure identification. Neural Networks 19(5), 623–643 (2006)
7. Kaburlasos, V.G., Petridis, V.: Fuzzy lattice neurocomputing (FLN) models. Neural Networks 13(10), 1145–1170 (2000)
8. Kaburlasos, V.G., Ritter, G.X. (eds.): Computational Intelligence Based on Lattice Theory. Studies in Computational Intelligence, vol. 67. Springer, Heidelberg (2007)
9. Kaburlasos, V. G., Athanasiadis, I. N., Mitkas, P. A.: Fuzzy lattice reasoning (FLR) classifier and its application for ambient ozone estimation. Intl J Approximate Reasoning (in press) (2007)
10. Kohonen, T.: Self-Organizing Maps. Information Sciences, vol. 30. Springer, Heidelberg (1995)
11. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. Intl. J Man-Machine Studies 7, 1–13 (1975)
12. Mendel, J.M.: Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice-Hall, Upper Saddle River, NJ (2001)
13. Pedrycz, W.: Knowledge-Based Clustering — From Data to Information Granules. John Wiley and Sons, Hoboken, NJ (2005)
14. Sugeno, M., Kang, G.T.: Structure identification of fuzzy model. Fuzzy Sets Systems 28(1), 15–33 (1988)
15. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. IEEE Trans Systems, Man, Cybern 15(1), 116–132 (1985)
16. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning, I. Information Sciences 8(3), 199–249 (1975)