Universal Approximation of a Class of Interval Type-2 Fuzzy Neural Networks Illustrated with the Case of Non-linear Identification

Juan R. Castro[1], Oscar Castillo[2], Patricia Melin[2], Antonio Rodriguez-Diaz[1], Olivia Mendoza[1]

[1]Baja California Autonomous University, UABC, Tijuana, B. C., Mexico.
2 Division of Graduate Studies and Research, Tijuana Institute of Technology, Tijuana, B. C., Mexico.
Email: jrcastror@uabc.mx, ocastillo@tectijuana.mx

Abstract—*Neural Networks (NN), Type-1 Fuzzy Logic Systems (T1FLS) and Interval Type-2 Fuzzy Logic Systems (IT2FLS) are universal approximators, they can approximate any non-linear function. Recent research shows that embedding T1FLS on an NN or embedding IT2FLS on an NN can be very effective for a wide number of non-linear complex systems, especially when handling imperfect information. In this paper we show that an Interval Type-2 Fuzzy Neural Network (IT2FNN) is a universal approximator with some precision using a set of rules and Interval Type-2 membership functions (IT2MF) and the Stone-Weierstrass Theorem. Also, simulation results of non-linear function identification using the IT2FNN for one and three variables with 10-fold cross-validation are presented.*

Keywords— Interval Type-2 Fuzzy Logic Systems, Interval Type-2 Fuzzy Neural Networks, Neural Networks, Universal Approximation.

1 Introduction

It has been shown that a three layer NN can approximate any real continuous function [1]. The same has been shown for a T1FLS [2, 3] using Stone-Weierstrass Theorem [6]. A similar analysis was made by Kosko [4, 5] using the concept of fuzzy regions. In [6, 7] Buckley shows that, with a Sugeno model [9], a T1FLS can be built with the ability to approximate any non-linear continuous function. Also, combining neural and fuzzy logic paradigms [10, 11], an effective tool can be created for approximating any non-linear function [8]. In this sense, an expert can use a Type-1 Fuzzy Neural Network (T1FNN) [13, 14] or IT2FNN systems and find interpretable solutions [12, 17-20]. In general, Takagi-Sugeno-Kang (TSK) T1FLS's approximate well by the use of polynomial consequent rules [9, 23].

This paper uses the Levenberg-Marquardt backpropagation learning algorithm for adapting antecedent and consequent parameters for an adaptive IT2FNN, since its efficiency and soundness characteristics made them fit for these optimizing problems.

An Adaptive IT2FNN is used as a universal approximator of any non-linear functions. A set Ψ of IT2FNN is universal if and only if (iff), given any process Ω, there is a system Φ ∈ Ψ such that the difference between the output from IT2FNN and output from Ω is less than a given ε.

2 Interval Type-2 Fuzzy Neural Networks

An IT2FNN [12] combines a TKS interval type-2 fuzzy inference system (TSKIT2FIS) [15,16] with an adaptive NN in order to take advantage of both models best characteristics. Even though Mamdami and Tsukamoto are much complex models [13, 14, 16], TSK is preferred due to the lower computational cost in type reduction.

2.1 IT2FNN-2:A2C0 Architecture

An IT2FNN-2:A2C0 (IT2FNN-0) is a seven layer IT2FNN, which integrates a first order TSKIT2FIS (interval type-2 fuzzy antecedents and real consequents), with an adaptive NN. The IT2FNN-2:A2C0 (Figure 1) layers are described as follows:

Layer 0: Inputs

$o_i^0 = x_i \quad ; i = 1,\ldots,n$

Layer 1: Adaptive type-1 fuzzy neuron (T1FN)

for i=1,…,n
  for k=s_i+1,…,s_{i+1}
    $net_k^1 = w_{k,i}^{1,0}\, x_i + b_k^1$
    $o_k^1 = \mu(net_k^1)$
  end
end

Layer 2: Non-adaptive T1FN

$o_{2k-1}^2 = o_{2k-1}^1 \cdot o_{2k}^1 \quad \forall\, k = 1,\ldots,\vartheta$

$o_{2k}^2 = o_{2k-1}^1 + o_{2k}^1 - o_{2k-1}^2$

for k=1,…,r
  for i=1,…,n
    $\tau = \mathrm{l}_{k,i} \quad ; \quad \pi = \sum_{j=1}^{i-1} v_j + |\tau|$
    if $\tau > 0$
      $\underline{\mu}_{k,i} = o_{i_l(\pi)}^2 \; ; \overline{\mu}_{k,i} = o_{i_u(\pi)}^2$
    else
      $\underline{\mu}_{k,i} = null \; ; \overline{\mu}_{k,i} = null$
    end
  end
end

Layer 3: Lower-upper firing strength rule normalization. Nodes in this layer are non-adaptive and the output is defined as a the ratio between the kth lower-upper firing strength rule and the sum of lower-upper firing strength of all rules:

ISBN: 978-989-95079-6-8

$$o_{2k-1}^3 = \underline{w}^k = \frac{\underline{f}^k}{\sum_{l=1}^{r} \underline{f}^l} \quad ; \quad k = 1,\dots,r$$

$$o_{2k}^3 = \overline{w}^k = \frac{\overline{f}^k}{\sum_{l=1}^{r} \overline{f}^l}$$

Layer 4: Rule consequents. Each node is adaptive and its parameters are $\{c_i^k, c_0^k\}$. The node's output corresponds to partial output of k^th rule y^k.

$$y^k = \sum_{i=1}^{n} c_i^k x_i + c_0^k \quad ; \quad k = 1,\dots,r$$

$$o_{2k-1}^4 = y^k$$

$$o_{2k}^4 = y^k$$

Layer 5: Type reduction. Estimates left-right interval values $[\hat{y}_l, \hat{y}_r]$, nodes are non-adaptive with outputs $\hat{y}_l, \hat{y}_r$. Layer 5 output is defined by:

$$o_1^5 = \hat{y}_l = \sum_{k=1}^{r} \underline{w}^k \cdot y^k$$

$$o_2^5 = \hat{y}_r = \sum_{k=1}^{r} \overline{w}^k \cdot y^k$$

Layer 6: Defuzzification. This layer's node is adaptive, where the output $\hat{y}$ is defined as weighted average of left-right values and parameter γ. Parameter γ (default value 0.5) adjusts the uncertainty interval defined by left-right values $[\hat{y}_l, \hat{y}_r]$.

$$o_1^6 = \hat{y} = \gamma \hat{y}_l + (1-\gamma)\hat{y}_r$$

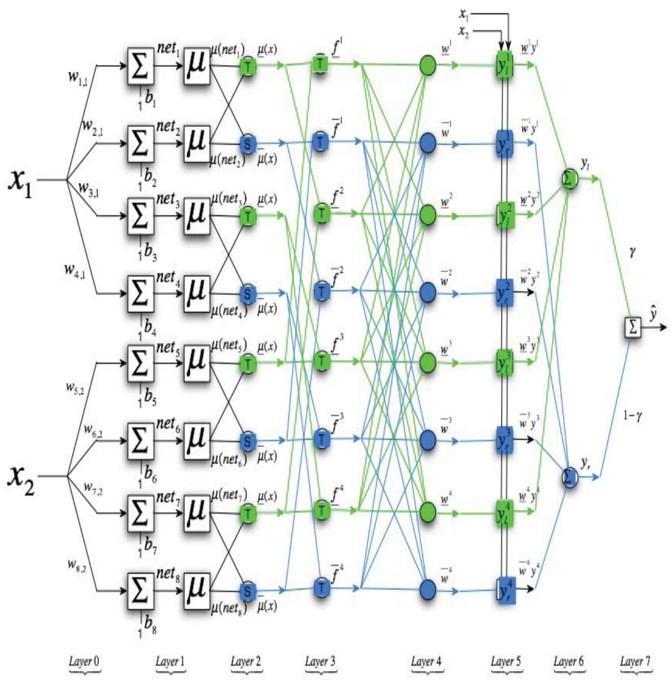

Figure 1: IT2FNN-2:A2C0 (IT2FNN-0) Architecture

2.2 *IT2FNN-2:A2C1 Architecture*

An IT2FNN-2:A2C1 (IT2FNN-2) [12] is a six layer IT2FNN, which integrates a first order TSKIT2FIS (interval type-2 fuzzy antecedents and interval type-1 fuzzy consequents), with an adaptive NN. The IT2FNN-2:A2C1 (Figure 2) layers are described as follows:

Layers 0-2 of IT2FNN-2:A2C1 architecture are the same as layers 0-2 of IT2FNN-2:A2C0 architecture.

Layer 3: Rule consequents. Each node is adaptive with parameters $\{c_i^k, c_0^k\}$ and $\{s_i^k, s_0^k\}$. Node's output corresponds to k^th rule's partial output, $\tilde{y}^k \in [y_l^k, y_r^k]$.

$$o_{2k-1}^3 = y_l^k = \sum_{i=1}^{n} c_i^k x_i + c_0^k - \sum_{i=1}^{n} s_i^k |x_i| - s_0^k \quad ; \quad k = 1,\dots,r$$

$$o_{2k}^3 = y_r^k = \sum_{i=1}^{n} c_i^k x_i + c_0^k + \sum_{i=1}^{n} s_i^k |x_i| + s_0^k$$

Layer 4: Type reduction. Estimates left-right values of interval $[\hat{y}_l, \hat{y}_r]$ and left(f_l^k)-right(f_r^k) firing strength that are used for computing $[\hat{y}_l, \hat{y}_r]$ using KM algorithm [15, 16].

$$f_l^k = leftpoint(\underline{f}^k, \overline{f}^k, y_l^k) \quad ; \quad f_r^k = rightpoint(\underline{f}^k, \overline{f}^k, y_r^k)$$

$$o_1^4 = \hat{y}_l = \frac{\sum_{k=1}^{r} f_l^k y_l^k}{\sum_{k=1}^{r} f_l^k} \quad ; \quad o_2^4 = \hat{y}_r = \frac{\sum_{k=1}^{r} f_r^k y_r^k}{\sum_{k=1}^{r} f_r^k}$$

Layer 5: Defuzzification. This layer's node is adaptive with output $\hat{y}$ defined by:

$$o_1^5 = \hat{y} = \gamma \hat{y}_l + (1-\gamma)\hat{y}_r$$

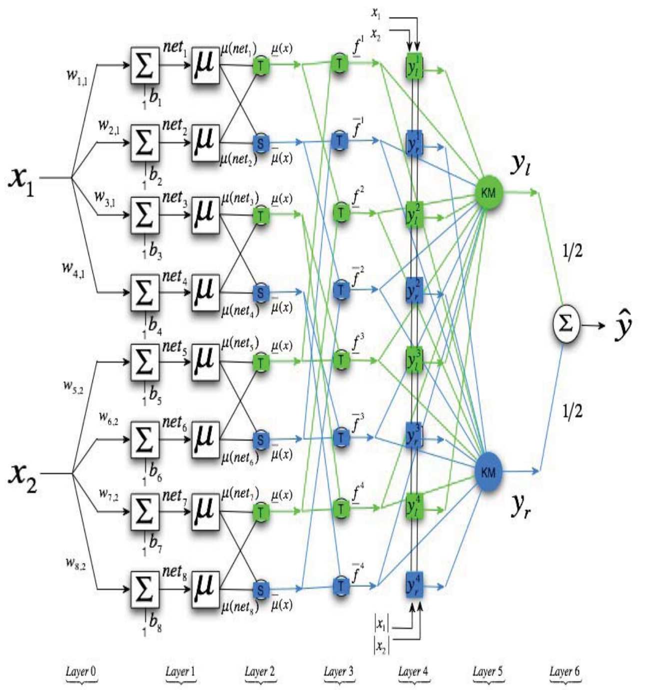

Figure 2: IT2FNN-2:A2C1 (IT2FNN-2) Architecture

ISBN: 978-989-95079-6-8

3 IT2FNN as a Universal Approximator

With this simplified interval type-2 fuzzy if-then rule, it is possible to prove that under certain circumstances, the resulting IT2FIS has unlimited approximation power to match any non-linear functions on a compact [24, 25] set using the Stone-Weierstrass theorem [13, 14, 21, 22] as follows:

3.1 Stone-Weierstrass Theorem.

Let the domain **D** be a compact space of **N** dimensions, and let $\Im$ be a set of continuous real-valued function on **D**, satisfying the following criteria [24, 25]:

1. **Identity function:** The constant $f(x) = 1$ is in $\Im$.
2. **Separability:** For any two points $x_1 \neq x_2$ in ***D***, there is an ***f*** in $\Im$ such that $f(x_1) \neq f(x_2)$.
3. **Algebraic closure:** If ***f*** and ***g*** are any two functions in $\Im$, then ***fg*** and ***af*** + ***bg*** are in $\Im$ for any two real numbers ***a*** and ***b***.

Then $\Im$ is dense in ***C***(***D***), the set of continuous real-value functions on **D**. In other words, for any $\varepsilon > 0$, and any function ***g*** in ***C***(***D***), there is a function ***f*** in $\Im$ such that $|g(x) - f(x)| < \varepsilon$ for all $\mathbf{x} \in \mathbf{D}$.

3.2 Applying Stone-Weierstrass Theorem used on IT2FNN-2:A2C0 Architecture.

In the IT2FNN, the domain in which we operate is almost always compact. It is a standard result in real analysis that every closed and bounded set in $\Re^n$ is compact. Now we shall apply the Stone-Weierstrass theorem to show the representational power of IT2FNN with simplified fuzzy if-then rules.

3.2.1 Identity Function

The first requirement of Stone-Weierstrass theorem needs our IT2FNN to be able to compute the identity function f(x)=1. An obvious way to compute the function is to set the consequence part of each rule equal to [1-δ, 1+δ], where δ is the uncertainty spread. In fact, an IT2FNN with only one rule suffices to satisfy this requirement.

3.2.2 Separability

The second requirement of the Stone-Weierstrass theorem needs the IT2FNN to be able to compute functions that have different values for different points. Without this requirement, the trivial set of functions f: f(x)= [c-δ, c+δ] c $\in \Re$ would satisfy Stone-Weierstrass theorem. Separability is satisfied whenever IT2FNN can compute strictly monotonic functions of each input variable. This can easily be achieved by adjusting the membership functions of the premise part.

3.2.3 Algebraic Closure-Additive

The third requirement of Stone-Weierstrass theorem needs the IT2FNN to be able to approximate sums and products of functions. Suppose we have two IT2FNNs S and $\overline{S}$ each of which has two rules. The output of each system can be expressed as

$$S : z = \gamma \cdot y_l + (1-\gamma) y_r = \gamma \frac{\underline{f}^1 y^1 + \underline{f}^2 y^2}{\underline{f}^1 + \underline{f}^2} + (1-\gamma) \frac{\overline{f}^1 y^1 + \overline{f}^2 y^2}{\overline{f}^1 + \overline{f}^2}$$

$$\overline{S} : \overline{z} = \overline{\gamma} \cdot \overline{y}_l + (1-\overline{\gamma}) \overline{y}_r = \overline{\gamma} \frac{\underline{g}^1 \overline{y}^1 + \underline{g}^2 \overline{y}^2}{\underline{g}^1 + \underline{g}^2} + (1-\overline{\gamma}) \frac{\overline{g}^1 \overline{y}^1 + \overline{g}^2 \overline{y}^2}{\overline{g}^1 + \overline{g}^2}$$

then the sum of z and $\overline{z}$ is equal to

$$az + b\overline{z} = \frac{\underline{f}^1 \overline{f}^1 \underline{g}^1 \overline{g}^1}{fg}\left[ay^1 + b\overline{y}^1\right] + \frac{\underline{f}^1 \overline{f}^1 \underline{g}^1 \overline{g}^2}{fg}\left[ay^1 + b\left(\overline{\gamma}\overline{y}^1 + (1-\overline{\gamma})\overline{y}^2\right)\right] +$$

$$\frac{\underline{f}^1 \overline{f}^1 \overline{g}^1 \underline{g}^2}{fg}\left[ay^1 + b\left(\overline{\gamma}\overline{y}^2 + (1-\overline{\gamma})\overline{y}^1\right)\right] + \frac{\underline{f}^1 \overline{f}^1 \underline{g}^2 \overline{g}^2}{fg}\left[ay^1 + b\overline{y}^2\right] +$$

$$\frac{\underline{f}^1 \overline{f}^2 \underline{g}^1 \overline{g}^1}{fg}\left[a\left(\gamma y^1 + (1-\gamma)y^2\right) + b\overline{y}^1\right] + \frac{\underline{f}^1 \overline{f}^2 \underline{g}^1 \overline{g}^2}{fg}\left[a\left(\gamma y^1 + (1-\gamma)y^2\right) + b\left(\overline{\gamma}\overline{y}^1 + (1-\overline{\gamma})\overline{y}^2\right)\right] +$$

$$\frac{\underline{f}^1 \overline{f}^2 \overline{g}^1 \underline{g}^2}{fg}\left[a\left(\gamma y^1 + (1-\gamma)y^2\right) + b\left(\overline{\gamma}\overline{y}^2 + (1-\overline{\gamma})\overline{y}^1\right)\right] + \frac{\underline{f}^1 \overline{f}^2 \underline{g}^2 \overline{g}^2}{fg}\left[a\left(\gamma y^1 + (1-\gamma)y^2\right) + b\overline{y}^2\right] +$$

$$\frac{\overline{f}^1 \underline{f}^2 \underline{g}^1 \overline{g}^1}{fg}\left[a\left(\gamma y^2 + (1-\gamma)y^1\right) + b\overline{y}^1\right] + \frac{\overline{f}^1 \underline{f}^2 \underline{g}^1 \overline{g}^2}{fg}\left[a\left(\gamma y^2 + (1-\gamma)y^1\right) + b\left(\overline{\gamma}\overline{y}^1 + (1-\overline{\gamma})\overline{y}^2\right)\right] +$$

$$\frac{\overline{f}^1 \underline{f}^2 \overline{g}^1 \underline{g}^2}{fg}\left[a\left(\gamma y^2 + (1-\gamma)y^1\right) + b\left(\overline{\gamma}\overline{y}^2 + (1-\overline{\gamma})\overline{y}^1\right)\right] + \frac{\overline{f}^1 \underline{f}^2 \underline{g}^2 \overline{g}^2}{fg}\left[a\left(\gamma y^2 + (1-\gamma)y^1\right) + b\overline{y}^2\right] +$$

$$\frac{\underline{f}^2 \overline{f}^2 \underline{g}^1 \overline{g}^1}{fg}\left[ay^2 + b\overline{y}^1\right] + \frac{\underline{f}^2 \overline{f}^2 \underline{g}^1 \overline{g}^2}{fg}\left[ay^2 + b\left(\overline{\gamma}\overline{y}^1 + (1-\overline{\gamma})\overline{y}^2\right)\right] +$$

$$\frac{\underline{f}^2 \overline{f}^2 \overline{g}^1 \underline{g}^2}{fg}\left[ay^2 + b\left(\overline{\gamma}\overline{y}^2 + (1-\overline{\gamma})\overline{y}^1\right)\right] + \frac{\underline{f}^2 \overline{f}^2 \underline{g}^2 \overline{g}^2}{fg}\left[ay^2 + b\overline{y}^2\right]$$

therefore we can construct an IT2FNN that computes $az + b\overline{z}$.

3.2.4 Algebraic Closure-Multiplicative

Modeling the product of $z\overline{z}$ of two IT2FNNs is the last capability we must demonstrate before we can conclude that the Stone-Weierstrass theorem can be applied to the proposed reasoning mechanism. The product $z\overline{z}$ can be expressed as

$$z\overline{z} = \frac{\underline{f}^1 \underline{g}^1 \left(\overline{f}^1 + \overline{f}^2\right)\left(\overline{g}^1 + \overline{g}^2\right)}{fg} \gamma y^1 \overline{\gamma}\overline{y}^1 + \frac{\underline{f}^1 \underline{g}^2 \left(\overline{f}^1 + \overline{f}^2\right)\left(\overline{g}^1 + \overline{g}^2\right)}{fg} \gamma y^1 \overline{\gamma}\overline{y}^2 +$$

$$\frac{\underline{f}^2 \underline{g}^1 \left(\overline{f}^1 + \overline{f}^2\right)\left(\overline{g}^1 + \overline{g}^2\right)}{fg} \gamma y^2 \overline{\gamma}\overline{y}^1 + \frac{\underline{f}^2 \underline{g}^2 \left(\overline{f}^1 + \overline{f}^2\right)\left(\overline{g}^1 + \overline{g}^2\right)}{fg} \gamma y^2 \overline{\gamma}\overline{y}^2 +$$

$$\frac{\underline{f}^1 \overline{g}^1 \left(\overline{f}^1 + \overline{f}^2\right)\left(\underline{g}^1 + \underline{g}^2\right)}{fg} \gamma y^1 (1-\overline{\gamma})\overline{y}^1 + \frac{\underline{f}^1 \overline{g}^2 \left(\overline{f}^1 + \overline{f}^2\right)\left(\underline{g}^1 + \underline{g}^2\right)}{fg} \gamma y^1 (1-\overline{\gamma})\overline{y}^2 +$$

ISBN: 978-989-95079-6-8

$$\frac{\underline{f}^2\overline{g}^1\left(\overline{f}^1+\overline{f}^2\right)\left(\underline{g}^1+\underline{g}^2\right)}{fg}\gamma y^2(1-\overline{\gamma})\overline{y}^1+\frac{\underline{f}^2\overline{g}^2\left(\overline{f}^1+\overline{f}^2\right)\left(\underline{g}^1+\underline{g}^2\right)}{fg}\gamma y^2(1-\overline{\gamma})\overline{y}^2+$$

$$\frac{\overline{f}^1\underline{g}^1\left(\underline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\overline{g}^2\right)}{fg}(1-\gamma)y^1\overline{\gamma}\overline{y}^1+\frac{\overline{f}^1\underline{g}^2\left(\underline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\overline{g}^2\right)}{fg}(1-\gamma)y^1\overline{\gamma}\overline{y}^2+$$

$$\frac{\overline{f}^2\underline{g}^1\left(\underline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\overline{g}^2\right)}{fg}(1-\gamma)y^2\overline{\gamma}\overline{y}^1+\frac{\overline{f}^2\underline{g}^2\left(\underline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\overline{g}^2\right)}{fg}(1-\gamma)y^2\overline{\gamma}\overline{y}^2+$$

$$\frac{\overline{f}^1\overline{g}^1\left(\underline{f}^1+\underline{f}^2\right)\left(\underline{g}^1+\underline{g}^2\right)}{fg}(1-\gamma)y^1(1-\overline{\gamma})\overline{y}^1+\frac{\overline{f}^1\overline{g}^2\left(\underline{f}^1+\underline{f}^2\right)\left(\underline{g}^1+\underline{g}^2\right)}{fg}(1-\gamma)y^1(1-\overline{\gamma})\overline{y}^2+$$

$$\frac{\overline{f}^2\overline{g}^1\left(\underline{f}^1+\underline{f}^2\right)\left(\underline{g}^1+\underline{g}^2\right)}{fg}(1-\gamma)y^2(1-\overline{\gamma})\overline{y}^1+\frac{\overline{f}^2\overline{g}^2\left(\underline{f}^1+\underline{f}^2\right)\left(\underline{g}^1+\underline{g}^2\right)}{fg}(1-\gamma)y^2(1-\overline{\gamma})\overline{y}^2$$

therefore we can construct an IT2FNN that computes $z\overline{z}$.

3.3 Applying Stone-Weierstrass theorem to the IT2FNN-2:A2C1 architecture

As in the case of IT2FNN-2:A2C0, let there be two distinct IT2FNN-2:A2C1, S and $\overline{S}$, each with two rules; we can proceed in this case as follows:

3.3.1 *Algebraic Closure-Additive*

$$S : z = \tfrac{1}{2}\left[\frac{\overline{f}^1 y_l^1+\underline{f}^2 y_l^2}{\overline{f}^1+\underline{f}^2}+\frac{\underline{f}^1 y_r^1+\overline{f}^2 y_r^2}{\underline{f}^1+\overline{f}^2}\right]$$

$$\overline{S} : \overline{z} = \tfrac{1}{2}\left[\frac{\overline{g}^1 \overline{y}_l^1+\underline{g}^2 \overline{y}_l^2}{\overline{g}^1+\underline{g}^2}+\frac{\underline{g}^1 \overline{y}_r^1+\overline{g}^2 \overline{y}_r^2}{\underline{g}^1+\overline{g}^2}\right]$$

then the sum of z and $\overline{z}$ is equal to

$$az+b\overline{z}=\frac{1}{2}\frac{\overline{f}^1\overline{g}^1}{\left(\overline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}\left(ay_l^1+b\overline{y}_l^1\right)+\frac{1}{2}\frac{\overline{f}^1\underline{g}^2}{\left(\overline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}\left(ay_l^1+b\overline{y}_l^2\right)+$$

$$\frac{1}{2}\frac{\underline{f}^2\overline{g}^1}{\left(\overline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}\left(ay_l^2+b\overline{y}_l^1\right)+\frac{1}{2}\frac{\underline{f}^2\underline{g}^2}{\left(\overline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}\left(ay_l^2+b\overline{y}_l^2\right)+$$

$$\frac{1}{2}\frac{\underline{f}^1\underline{g}^1}{\left(\underline{f}^1+\overline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}\left(ay_r^1+b\overline{y}_r^1\right)+\frac{1}{2}\frac{\underline{f}^1\overline{g}^2}{\left(\underline{f}^1+\overline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}\left(ay_r^1+b\overline{y}_r^2\right)+$$

$$\frac{1}{2}\frac{\overline{f}^2\underline{g}^1}{\left(\underline{f}^1+\overline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}\left(ay_r^2+b\overline{y}_r^1\right)+\frac{1}{2}\frac{\overline{f}^2\overline{g}^2}{\left(\underline{f}^1+\overline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}\left(ay_r^2+b\overline{y}_r^2\right)$$

therefore we can construct an IT2FNN that computes $az+b\overline{z}$.

3.3.2 *Algebraic Closure-Multiplicative*

The product $z\overline{z}$ can be expressed as

$$z\overline{z}=\frac{1}{4}\frac{\overline{f}^1\overline{g}^1}{\left(\overline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}y_l^1\overline{y}_l^1+\frac{1}{4}\frac{\overline{f}^1\underline{g}^2}{\left(\overline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}y_l^1\overline{y}_l^2+$$

$$\frac{1}{4}\frac{\underline{f}^2\overline{g}^1}{\left(\overline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}y_l^2\overline{y}_l^1+\frac{1}{4}\frac{\underline{f}^2\underline{g}^2}{\left(\overline{f}^1+\underline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}y_l^2\overline{y}_l^2+$$

$$\frac{1}{4}\frac{\overline{f}^1\underline{g}^1}{\left(\overline{f}^1+\underline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}y_l^1\overline{y}_r^1+\frac{1}{4}\frac{\overline{f}^1\overline{g}^2}{\left(\overline{f}^1+\underline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}y_l^1\overline{y}_r^2+$$

$$\frac{1}{4}\frac{\underline{f}^2\underline{g}^1}{\left(\overline{f}^1+\underline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}y_l^2\overline{y}_r^1+\frac{1}{4}\frac{\underline{f}^2\overline{g}^2}{\left(\overline{f}^1+\underline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}y_l^2\overline{y}_r^2+$$

$$\frac{1}{4}\frac{\underline{f}^1\overline{g}^1}{\left(\underline{f}^1+\overline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}y_r^1\overline{y}_l^1+\frac{1}{4}\frac{\underline{f}^1\underline{g}^2}{\left(\underline{f}^1+\overline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}y_r^1\overline{y}_l^2+$$

$$\frac{1}{4}\frac{\overline{f}^2\overline{g}^1}{\left(\underline{f}^1+\overline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}y_r^2\overline{y}_l^1+\frac{1}{4}\frac{\overline{f}^2\underline{g}^2}{\left(\underline{f}^1+\overline{f}^2\right)\left(\overline{g}^1+\underline{g}^2\right)}y_r^2\overline{y}_l^2+$$

$$\frac{1}{4}\frac{\underline{f}^1\underline{g}^1}{\left(\underline{f}^1+\overline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}y_r^1\overline{y}_r^1+\frac{1}{4}\frac{\underline{f}^1\overline{g}^2}{\left(\underline{f}^1+\overline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}y_r^1\overline{y}_r^2+$$

$$\frac{1}{4}\frac{\overline{f}^2\underline{g}^1}{\left(\underline{f}^1+\overline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}y_r^2\overline{y}_r^1+\frac{1}{4}\frac{\overline{f}^2\overline{g}^2}{\left(\underline{f}^1+\overline{f}^2\right)\left(\underline{g}^1+\overline{g}^2\right)}y_r^2\overline{y}_r^2$$

therefore we can construct an IT2FNN that computes $z\overline{z}$.
The IT2FNN architectures that compute $az+b\overline{z}$ and $z\overline{z}$ have the same S and $\overline{S}$ if and only if the class of membership functions is invariant under multiplication. This is loosely true if the class of interval type-2 membership functions is the set of all bell-shaped functions and Gaussian-shaped (igbellmtype2, igbellstype2 and igbelltype2 for bell-shaped and igaussmtype2, igaussstype2 and igausstype2 for Gaussian-shaped, all of which can be found in the Matlab's Interval Type-2 Fuzzy Logic Toolbox [26]).

Therefore by choosing an appropriate class of membership functions, we can conclude that the IT2FNN with simplified fuzzy if-then rules satisfy the four criteria of Stone-Weierstrass theorem. Consequently, for any given ε > 0 and any real-valued function g, there is an IT2FNN S such that $\left|g(x)-S(x)\right|<\varepsilon$ for all x in underlying compact set. Moreover since the simplified IT2FNN is a proper subset of two types of architecture, we can conclude that the IT2FNN architecture has unlimited approximation power to match any given data set.

4 Application Examples

Two application examples are used to illustrate the proofs of universality, as follows.

Experiment 1. Identification of a one variable non-linear function.

In this experiment we approximate a non-linear function $f:\Re\rightarrow\Re$:

$$f(u) = 0.6\sin(\pi u) + 0.3\sin(3\pi u) + 0.1\sin(5\pi u) + \eta$$

(where η is a uniform noise component) using a one input-one output IT2FNN, 50 training data sets with 10 fold cross-validation with uniform noise levels, 6 IT2MF type igaussmtype2, 6 rules and 50 epochs. Once ANFIS and IT2FNN models are identified a comparison was made, taking into account RMSE statistic values with 10 fold cross-validation. Table 1 and Figure 3 show resulting RMSE values for ANFIS and IT2FNN, it can be seen that IT2FNN architectures [12] perform better than ANFIS.

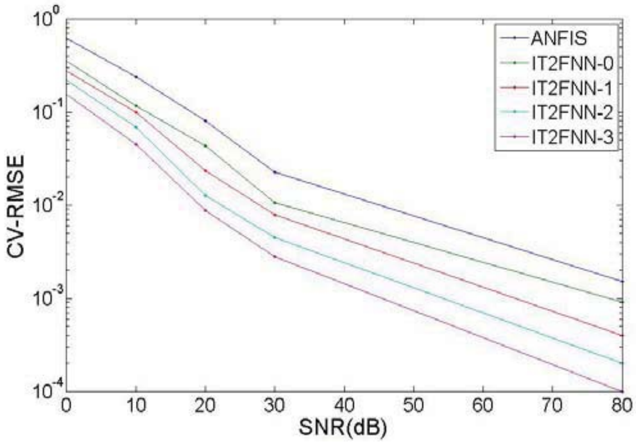

Figure 3: RMSE values of ANFIS and IT2FNN using 10-fold cross-validation for identifying non-linearity in Experiment 1.

Table 1: RMSE values of ANFIS and IT2FNN with 10-fold cross-validation for identifying non-linearity of Experiment 1.

SNR(dB)	ANFIS	IT2FNN-0	IT2FNN-1	IT2FNN-2	IT2FNN-3
0	0.6156	0.3532	0.2764	0.2197	0.1535
10	0.2375	0.1153	0.0986	0.0683	0.0453
20	0.0806	0.0435	0.0234	0.0127	0.0087
30	0.0225	0.0106	0.0079	0.0045	0.0028
free	0.0015	0.0009	0.0004	0.0002	0.0001

Experiment 2. Identification of a three variable non-linear function.

A three input – one output IT2FNN is used to approximate non-linear Sugeno [9] function $f:\Re^3 \rightarrow \Re$:

$$f(x_1, x_2, x_3) = \left(1 + \sqrt{x_1} + \frac{1}{x_2} + \frac{1}{\sqrt{x_3^3}}\right)^2 + \eta$$

216 training data sets are generated with 10-fold cross-validation and 125 for tests; 2 igaussmtype2 IT2MF for each input, 8 rules and 50 epochs. Once the ANFIS and IT2FNN models are identified, a comparison is made with RMSE statistic values and 10-fold cross-validation. Table 2 and Figure 4 show the resultant RMSE values for ANFIS and IT2FNN. It can be seen that IT2FNN architectures [12] perform better than ANFIS.

Table 2: Resulting RMSE values in ANFIS and IT2FNN for non-linearity identification in Experiment 2 with 10-fold cross-validation.

SNR(dB)	ANFIS	IT2FNN-0	IT2FNN-1	IT2FNN-2	IT2FNN-3
0	1.0432	0.7203	0.6523	0.5512	0.5267
10	0.3096	0.2798	0.2583	0.2464	0.2344
20	0.1703	0.1637	0.1592	0.1465	0.1387
30	0.1526	0.1408	0.1368	0.1323	0.1312
free	0.1503	0.1390	0.1323	0.1304	0.1276

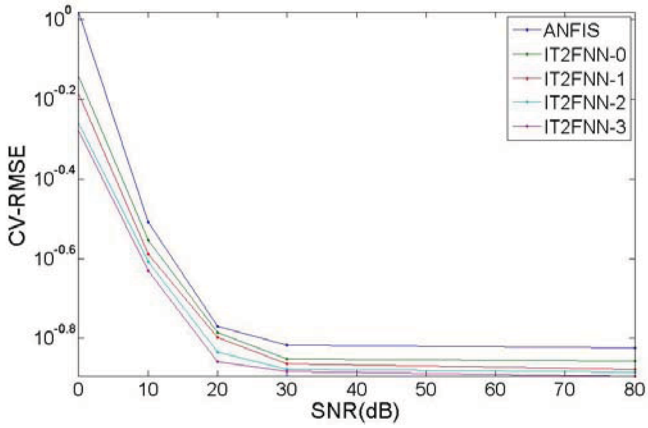

Figure 4: Resulting RMSE values obtained by ANFIS and IT2FNN for non-linearity identification in Experiment 2 with 10-fold cross-validation.

5 Conclusions

In these experiments, the estimated RMSE values for non-linear function identification with 10-fold cross-validation for the hybrid architectures IT2FNN-2:A2C0 and IT2FNN-2:A2C1 illustrate the proof based on Stone-Weierstrass theorem that they are universal approximators for efficient identification of non-linear functions. Also, it can be seen that while increasing the Signal Noise Ratio (SNR), IT2FNN architectures handle uncertainty more efficiently.

Acknowledgment

The authors would like to thank CONACYT and DGEST for the financial support given for this research project. The student (Juan R. Castro) is supported by a scholarship from UABC-CONACYT.

References

[1] K. Hornick, M., Stinchcombe and H. White, Multilayer feedforward networks are universal approximators. Neural Networks, 2(5):359-366, 1989.

[2] L. Wang, and J. Mendel, Fuzzy basis functions, universal approximation, and orthogonal least squares learning. IEEE Trans. Neural Networks, 3:807-813, 1992.

[3] L. Wang and J. Mendel, Generating fuzzy rules by learning from examples. IEEE Trans. Syst. Man and Cyber., 22(6):1414-1427, 1992.

[4] B. Kosko, Neural Networks and Fuzzy Systems. Englewood Cliffs, NJ, Prentice Hall, 1992.

[5] B. Kosko, Fuzzy systems as universal approximators. IEEE Trans. Computers, 43(1):1329-1333, 1994.

[6] J. Buckley, Universal fuzzy controllers. Automatica, 28(6):1245-1248, 1992.

[7] J. Buckley, Sugeno type controllers are universal controllers. Automatica, 53:299-304, 1993.

[8] J. Buckley and Y. Hayashi, Can fuzzy neural networks approximate continuos fuzzy functions?, Fuzzy Sets and Systems, 61:43-52, 1994.

[9] T. Takagi and M. Sugeno, Fuzzy Identification of Systems and its applications to modelling and Control, *IEEE Transactions on Systems, Man, and Cybernetics*, 15(1): 116-132, Jan-Feb 1985.

[10] Lotfi A. Zadeh. Fuzzy sets. *Information and Control*, 8:338-353, 1965.

[11] K. Hirota and W. Pedrycz, OR/AND neuron in modeling fuzzy set connectives, IEEE Transactions on Fuzzy Systems, 2(994) 151-161.

[12] J.R. Castro, O. Castillo, P. Melin, A. Rodriguez-Diaz, A Hybrid Learning Algorithm for Interval Type-2 Fuzzy Neural Networks: The Case of Time Series Prediction; In Soft Computing for Hybrid Intelligent Systems. Book Chapter, Volume 154/2008: 363-386, Springer Berlin / Heidelberg. ISBN: 978-3-540-70811-7, ISSN: 1860-949X (Print) 1860-9503 (Online), September 2008.

[13] J.-S.R. Jang, "ANFIS: adaptive-network-based fuzzy inference system". IEEE Trans. Syst., Man, Cybern., vol. 23, no. 3, pp. 665-684, 1993.

[14] J.-S.R. Jang, C.T. Sun, and E. Mizutani, Neuro-fuzzy and Soft Computing. New York: Prentice-Hall, 1997.

[15] Q. Liang and J.M. Mendel, "Interval type-2 logic systems: theory and design," IEEE Transactions on Fuzzy Systems, vol. 8 no. 5, pp. 535-550, Oct. 2000.

[16] J.M. Mendel, Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions, Prentice Hall, NJ. 2001.

[17] Lee, C-H., Hu, T-W, Lee, C-T, Lee, Y-C, "A Recurrent Interval Type-2 Fuzzy Neural Network with Asymmetric Membership Functions for Nonlinear System Identification". In FUZZ-IEEE 2008, Hong Kong, June 2008.

[18] J.R. Castro, O. Castillo, P. Melin, A. Rodriguez-Diaz, Hybrid learning algorithm for interval type-2 fuzzy neural networks, In Proceedings of Granular Computing 2007, pages 157-162, Silicon Valley, CA, USA, ISBN: 0-7695-3032-X, November 2007.

[19] C.H. Lee, J.L. Hong, Y.C. Lin, and W.Y. Lai, "Type-2 Fuzzy Neural Network Systems and Learning," International Journal of Computational Cognition, Vol. 1, No. 4, pp. 79-90, 2003.

[20] C.H. Wang, C.S. Cheng, and Tsu-Tian Lee, "Dynamical optimal training for interval type-2 fuzzy neural network (T2FNN)," IEEE Trans. on Systems, Man, and Cybernetics Part B: Cybernetics, Vol. 34, No. 3, pp. 1462-1477, 2004.

[21] L.V. Kantorovich and G.P. Akilov. Functional analysis. Pergamon, Oxford, 2nd edition, 1982.

[22] H.L. Royden. Real analysis. Macmillan, New York, 2nd edition 1968.

[23] V. Kreinovich, G.C. Mouzouris and H.T. Nguyen, "Fuzzy Rule Based Modelling as a Universal Aproximation Tool," in Fuzzy Systems, Modeling and Control (H. T. Nguyen and M. Sugeno, Eds.), pp. 135-195, Kluwer Ac. Publ., Boston, 1998.

[24] C.T. Scarborough, A.H. Stone, Products of nearly compact spaces, Transactions of the American Mathematical Society 124: 131-147, 1966.

[25] R.E. Edwards, Functional Analysis: Theory and Applications, Dover Publications, 1995.

[26] J.R. Castro, O. Castillo, P. Melin, A. Rodriguez-Diaz, Building fuzzy inference systems with a new interval type-2 fuzzy logic toolbox, In Transactions on Computational Science. Book Chapter, Volume 4750:104-114, Springer-Verlag. ISBN: 978-3-540-79298-7, ISSN: 0302-9743 (Print) 1611-3349 (Online), June 2008.