Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2017, Article ID 8594738, 10 pages
https://doi.org/10.1155/2017/8594738

Research Article

Fuzzy Modeling for Uncertainty Nonlinear Systems with Fuzzy Equations

Raheleh Jafari and Wen Yu

Departamento de Control Automatico, CINVESTAV-IPN (National Polytechnic Institute), Mexico City, Mexico

Correspondence should be addressed to Wen Yu; yuw@ctrl.cinvestav.mx

Received 19 July 2016; Revised 26 October 2016; Accepted 21 December 2016; Published 22 January 2017

Academic Editor: Rosana Rodriguez-Lopez

The uncertain nonlinear systems can be modeled with fuzzy equations by incorporating the fuzzy set theory. In this paper, the fuzzy equations are applied as the models for the uncertain nonlinear systems. The nonlinear modeling process is to find the coefficients of the fuzzy equations. We use the neural networks to approximate the coefficients of the fuzzy equations. The approximation theory for crisp models is extended into the fuzzy equation model. The upper bounds of the modeling errors are estimated. Numerical experiments along with comparisons demonstrate the excellent behavior of the proposed method.

1. Introduction

Since the uncertainties in the nonlinear systems can be transformed into fuzzy set theory [1], the fuzzy systems are good models for uncertainty systems. The fuzzy models are based on fuzzy rules. These fuzzy rules give information of the uncertain nonlinear systems. Any nonlinear system can be approximated by several piecewise linear systems (Takagi-Sugeno fuzzy model [2]) or known nonlinear systems (Mamdani fuzzy model [3]). The uncertain nonlinear systems can be modeled by the fuzzy models with simple linear or nonlinear models.

The nonlinear systems can be also modeled with difference equations and algebraic systems. Interpolation methodology has been broadly utilized for function approximation as well as system identification [4, 5]. In [6], the fuzzy polynomial interpolation is applied for system modeling. The coefficients of the polynomials are the fuzzy numbers [7], such that the uncertainties are interpolated with the fuzzy set theory. The theory problem associated with polynomial interpolation is discussed in [8]. It concludes that the interpolation of the function $f(x)$ includes $O(n)$ time complexity at n data points.

In [9], two-dimensional polynomial interpolation is demonstrated. The constraint associated with multivariable interpolation has been investigated in [6], where the Newton-form interpolation is employed. In [10], the multivariate Vandermonde matrix is utilized. Smooth function approximation has been broadly implemented currently [11, 12]. It yields a model by utilizing Lagrange interpolating polynomials at the points of product grids [7, 13]. However if it involves uncertainties in the interpolation points, the above suggested techniques will not work appropriately.

The fuzzy equation can be regarded as a generalized form of the fuzzy polynomial. Compared with the normal fuzzy systems, the fuzzy equations are more easy to be applied, because the uncertainties are direct fuzzy parameters of the fuzzy equations. However, these parameters are not easy to be obtained. There are several approaches to construct the fuzzy equations. Reference [14] utilized the parametric form of fuzzy numbers and restored the original fuzzy equations using crisp linear systems. In [15], the extension principle is implemented and it suggests that the coefficients can be either real or complex fuzzy numbers. Whatsoever, the validation of the solution is not assured. Reference [16] prescribed the homeotypic analysis methodology. Reference [17] inducted Newton's technique. In [18], the solution of fuzzy equations is extracted using the fixed point methodology. One of the well-known methods is termed as α-level [19]. By using the method of superimposition of sets, fuzzy numbers can be solved. Recently, fuzzy fractional differential and integral equations have been extensively studied in [20–22]. However, the above methods are very complex.

The numerical solution associated with fuzzy equation can be fetched using the iterative technique [23], interpolation technique [24], and Runge-Kutta technique [25]. It can also be implemented to fuzzy differential equations [26]. These methods are also difficult to be applied. Both neural networks as well as fuzzy logic are considered to be the universal estimators which can estimate any nonlinear function to any notified precision [27]. Current outcomes demonstrate that the fusion methodology of these two different techniques appears to be highly efficient for nonlinear systems identification [28]. Neural networks can also be implemented for resolving the fuzzy equation. In [29], the simple fuzzy quadratic equation is resolved by the neural network method. References [30, 31] elaborated the outcomes of [29] into fuzzy polynomial equation. In [32, 33], the solution of dual fuzzy equations is obtained by neural networks. A matrix pattern associated with the neural learning has been quoted in [34]. However, these techniques are not general; they cannot give the fuzzy coefficients directly with neural networks [35, 36].

We use the neural network method to approximate the coefficients of the fuzzy equations as in our previous paper [37]. In this paper, the standard backpropagation method is modified, such that the fuzzy numbers in the fuzzy ideations can be trained. The approximation theory of the crisp models is extended into the fuzzy equations. The upper bounds of the modeling errors with fuzzy equations are estimated. Finally, we use some real examples to show the effectiveness of our approximation method.

2. Nonlinear System Modeling with Fuzzy Equations

A general discrete-time nonlinear system can be described as

$$\begin{aligned}\overline{w}_{r+1} &= \overline{\rho}\left[\overline{w}_r, v_r\right],\\ z_r &= \overline{\varphi}\left[\overline{w}_r\right].\end{aligned} \tag{1}$$

Here we consider $v_r \in \mathfrak{R}^u$ as the input vector, $\overline{w}_r \in \mathfrak{R}^l$ is regarded as an internal state vector, and $z_r \in \mathfrak{R}^m$ is the output vector. $\overline{\rho}$ and $\overline{\varphi}$ are noted as generalized nonlinear smooth functions $\overline{\rho}, \overline{\varphi} \in C^{\infty}$. Define $Z_r = [z_{r+1}^T, z_r^T, \ldots]^T$ and $V_r = [v_{r+1}^T, v_r^T, \ldots]^T$. Supposing $\partial Z/\partial\overline{w}$ is nonsingular at the instance $\overline{w}_r = 0$, $V_r = 0$, this will create a path towards the following model:

$$z_r = \Omega\left[z_{r-1}^T, z_{r-2}^T, \ldots, v_r^T, v_{r-1}^T, \ldots\right], \tag{2}$$

where $\Omega(\cdot)$ is a nonlinear difference equation exhibiting the plant dynamics and v_r and z_r are computable scalar input and output, respectively. The nonlinear system which is represented by (2) is implied as a NARMA model. The input of the system with incorporated nonlinearity is considered to be

$$w_r = \left[z_{r-1}^T, z_{r-2}^T, \ldots, v_r^T, v_{r-1}^T, \ldots\right]^T. \tag{3}$$

Taking into consideration the nonlinear systems as mentioned in (2), it can be simplified as the following linear-in-parameter model:

$$z_r = \sum_{i=0}^{n} c_i w_r^i, \tag{4}$$

where c_i is considered to be the linear parameter and w_r^i is nonlinear function. The variables related to this function are quantifying input and output.

Many nonlinear systems can be expressed by linear-in-parameter models such as Lagrangian mechanical systems. The parameters of these models are uncertain and the uncertainties satisfy the fuzzy set theory [1]. In this way, the inconvenience problems in nonlinear modeling such as complexity and uncertainty are solved by the fuzzy logic theory and linear-in-parameter structure. The modeling process with the fuzzy equation is to find the fuzzy coefficients of the linear-in-parameter model such that the fuzzy equation can represent the uncertain nonlinear system.

We assume that the model of the nonlinear system (4) has uncertainties in the parameter c_i. The following definitions will be used in this paper.

Definition 1 (fuzzy number). A fuzzy number A is a function $A \in E : \mathfrak{R} \rightarrow [0,1]$; in such a way, (1) A is normal, (there prevail $w_0 \in \mathfrak{R}$ in such a way that $A(w_0) = 1$); (2) A is convex, $A(\lambda w + (1-\lambda)z) \geq \min\{A(w), A(z)\}$, $\forall w, z \in \mathfrak{R}$, $\forall \lambda \in [0,1]$; (3) A is upper semicontinuous on $\mathfrak{R}$; that is, $A(w) \leq A(w_0) + \varepsilon$, $\forall w \in N(w_0)$, $\forall w_0 \in \mathfrak{R}$, $\forall \varepsilon > 0$, $N(w_0)$ is a neighborhood; (4) the set $A^+ = \{w \in \mathfrak{R},\ A(w) > 0\}$ is compact.

In order to demonstrate the fuzzy numbers, the membership functions are utilized. The most widely discussed membership functions are noted to be the triangular function

$$\mu_A = F(a,b,c) = \begin{cases} \dfrac{w-a}{b-a} & a \leq w \leq b \\ \dfrac{c-w}{c-b} & b \leq w \leq c \end{cases} \tag{5}$$
$$\text{otherwise } \mu_A = 0$$

and trapezoidal function

$$\mu_A = F(a,b,c,d) = \begin{cases} \dfrac{w-a}{b-a} & a \leq w \leq b \\ \dfrac{d-w}{d-c} & c \leq w \leq d \\ 1 & b \leq w \leq c \end{cases} \tag{6}$$
$$\text{otherwise } \mu_A = 0.$$

On a par with crisp variable, the fuzzy variable A possesses three essential operations: $\oplus$, $\ominus$, and $\odot$. They signify these operations: sum, subtract, and multiply.

The fuzzy variable A which contains the dimension of w is dependent on the membership function, (5) has three variables, and (6) has four variables. In order to define consistency operations, we first apply α-level operation to the fuzzy number.

Definition 2 (α-level). The α-level associated with a fuzzy number A is stated as

$$[A]^{\alpha} = \{w \in \Re,\ A(w) \geq 0\}, \tag{7}$$

where $0 < \alpha \leq 1$, $A \in E$.

Therefore $[A]^0 = A^+ = \{w \in \Re,\ A(w) > 0\}$. Since $\alpha \in [0,1]$, $[A]^{\alpha}$ is bounded mentioned as $\underline{A}^{\alpha} \leq [A]^{\alpha} \leq \overline{A}^{\alpha}$. The α-level of A between $\underline{A}^{\alpha}$ and $\overline{A}^{\alpha}$ is explained as

$$[A]^{\alpha} = \Phi\left(\underline{A}^{\alpha}, \overline{A}^{\alpha}\right). \tag{8}$$

Definition 3 (fuzzy operations). If $A_1, A_2 \in E$, $\lambda \in \mathbb{R}$, the fuzzy operations are as follows:

Sum:

$$\begin{aligned}[A_1 \oplus A_2]^{\alpha} &= [A_1]^{\alpha} + [A_2]^{\alpha} \\ &= \left[\underline{A_1}^{\alpha} + \underline{A_2}^{\alpha}, \overline{A_1}^{\alpha} + \overline{A_2}^{\alpha}\right].\end{aligned} \tag{9}$$

Subtract:

$$\begin{aligned}[A_1 \ominus A_2]^{\alpha} &= [A_1]^{\alpha} - [A_2]^{\alpha} \\ &= \left[\underline{A_1}^{\alpha} - \underline{A_2}^{\alpha}, \overline{A_1}^{\alpha} - \overline{A_2}^{\alpha}\right].\end{aligned} \tag{10}$$

Multiply:

$$\begin{aligned}&(A_1 \odot A_2)^{\alpha} \\ &\quad = \Phi\left(\min\left\{\underline{A_1}^{\alpha}\underline{A_2}^{\alpha}, \underline{A_1}^{\alpha}\overline{A_2}^{\alpha}, \overline{A_1}^{\alpha}\underline{A_2}^{\alpha}, \overline{A_1}^{\alpha}\overline{A_2}^{\alpha}\right\},\right. \\ &\quad \left.\max\left\{\underline{A_1}^{\alpha}\underline{A_2}^{\alpha}, \underline{A_1}^{\alpha}\overline{A_2}^{\alpha}, \overline{A_1}^{\alpha}\underline{A_2}^{\alpha}, \overline{A_1}^{\alpha}\overline{A_2}^{\alpha}\right\}\right).\end{aligned} \tag{11}$$

Scalar multiplication: $\alpha \in [0,1]$.

$$[\lambda A]^{\alpha} = \lambda [A]^{\alpha} = \begin{cases}\Phi\left(\lambda\underline{A}^{\alpha}, \lambda\overline{A}^{\alpha}\right) & \lambda \geq 0 \\ \Phi\left(\lambda\overline{A}^{\alpha}, \lambda\underline{A}^{\alpha}\right) & \lambda < 0.\end{cases} \tag{12}$$

Definition 4 (absolute value). Absolute value of a triangular fuzzy number $A(w) = F(a,b,c)$ is

$$|A(w)| = |a| + |b| + |c|. \tag{13}$$

Now we utilize the following fuzzy equation to model the uncertain nonlinear system (1):

$$z_r = c_1 f_1(w_r) \oplus c_2 f_2(w_r) \oplus \cdots \oplus c_n f_n(w_r). \tag{14}$$

Because c_i is fuzzy number, we apply the fuzzy operation $\oplus$.

Taking into consideration a particular case, $f_i(w_r)$ has polynomial form,

$$z_r = c_1 w_r \oplus \cdots \oplus c_n w_r^n. \tag{15}$$

Equation (15) is termed as fuzzy polynomial.

Modeling with fuzzy equation (or fuzzy polynomial) can be regarded as fuzzy interpolation. In this paper, we utilize the fuzzy equation (14) to model the uncertain nonlinear system (1), in such a manner that the output related to the plant z_r can approach the desired output z_r^*,

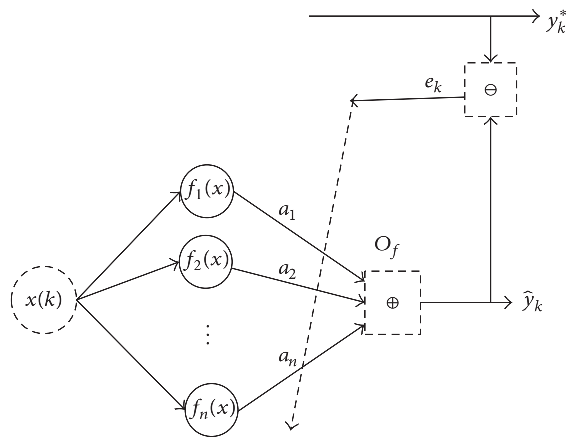


Figure 1: Fuzzy equation in the form of a neural network.

$$\min_{c_r} \left\| z_r - z_r^* \right\|. \tag{16}$$

This modeling object can be regarded as a way to detect c_r for the following fuzzy equation:

$$z_r^* = c_1 f_1(w_r) \oplus c_2 f_2(w_r) \oplus \cdots \oplus c_n f_n(w_r), \tag{17}$$

where $w_r = [z_{r-1}^T, z_{r-2}^T, \ldots, v_r^T, v_{r-1}^T, \ldots]^T$.

3. Fuzzy Parameter Estimation with Neural Networks

We design a neural network to represent the fuzzy equation (14); see Figure 1. The input to the neural network is w_r and the output is the fuzzy number z_r. The weights are c_i. The main idea is to detect appropriate weight of neural network c_i in such a manner that the output of the neural network z_r converges to the desired output z_r^*.

In order to simplify the operation of the neural network, we use the triangular fuzzy number (5) in this paper. The input fuzzy number w_r is first applied to α-level as in (7):

$$[w_r]^{\alpha} = \Phi\left(\underline{w_r}^{\alpha}, \overline{w_r}^{\alpha}\right) \quad r = 0 \cdots n. \tag{18}$$

Then we have

$$[O_i]^{\alpha} = \Phi f_i\underline{(w_r)}^{\alpha}, f_i\overline{(w_r)}^{a} \quad i = 1 \cdots n. \tag{19}$$

The neural network output is

$$\begin{aligned}[z_r]^{\alpha} = \Phi &\left\{\sum_{i \in M}\underline{O_i}^{\alpha}\underline{c_i}^{\alpha} + \sum_{i \in C}\overline{O_i}^{\alpha}\underline{c_i}^{\alpha} + \underline{c_0}^{\alpha}, \sum_{i \in M'}\overline{O_i}^{\alpha}\overline{c_i}^{\alpha}\right. \\ &\left. + \sum_{i \in C'}\underline{O_i}^{\alpha}\overline{c_i}^{\alpha} + \overline{c_0}^{\alpha}\right\},\end{aligned} \tag{20}$$

where $M = \{i \mid \underline{c_i}^{\alpha} \geq 0\}$, $C = \{i \mid \underline{c_i}^{\alpha} < 0\}$, $M' = \{i \mid \overline{c_i}^{\alpha} \geq 0\}$, and $C' = \{i \mid \overline{c_i}^{\alpha} < 0\}$.

In order to train the weights, we need to define a cost function for the fuzzy numbers. The error of the training is

$$e_r = z_r^* - z_r, \tag{21}$$

where $[z_r^*]^\alpha = \Phi(\underline{z_r^*}^\alpha, \overline{z_r^*}^\alpha)$, $[z_r]^\alpha = \Phi(\underline{z_r}^\alpha, \overline{z_r}^\alpha)$, and $[e_r]^\alpha = \Phi(\underline{e_r}^\alpha, \overline{e_r}^\alpha)$. The cost function is defined as

$$\begin{aligned} \Psi_r &= \underline{\Psi}^\alpha + \overline{\Psi}^\alpha, \\ \underline{\Psi}^\alpha &= \frac{1}{2}\left(\underline{z_r^*}^\alpha - \underline{z_r}^\alpha\right)^2, \\ \overline{\Psi}^\alpha &= \frac{1}{2}\left(\overline{z_r^*}^\alpha - \overline{z_r}^\alpha\right)^2. \end{aligned} \tag{22}$$

Ψ_r is considered to be a scalar function. It is quite obvious that $\Psi_r \to 0$ means $[z_r]^\alpha \to [z_r^*]^\alpha$.

The vital positiveness that lies within the least mean square (22) is that it has a self-correcting feature that makes it suitable to function for arbitrarily vast duration without shifting from its constraints. The mentioned gradient algorithm is subjected to cumulative series of errors and is convenient for long runs in absence of an additional error rectification procedure. It is more robust in statistics, identification, and signal processing [38].

Now we use gradient method to train the weight $c_i = [c_i^1, c_i^2, c_i^3]$ defined in (5), or c_i^k, $k = 1, 2, 3$. We compute $\partial\Psi_r/\partial c_i^k$ as

$$\frac{\partial \Psi_r}{\partial c_i^k} = \frac{\partial \underline{\Psi}^\alpha}{\partial c_i^k} + \frac{\partial \overline{\Psi}^\alpha}{\partial c_i^k}. \tag{23}$$

According to the chain rule

$$\frac{\partial \underline{\Psi}^\alpha}{\partial c_i^k} = \frac{\partial \underline{\Psi}^\alpha}{\partial \underline{z_r}^\alpha}\frac{\partial \underline{z_r}^\alpha}{\partial \underline{c_i}^\alpha}\frac{\partial \underline{c_i}^\alpha}{\partial c_i^k} = \left(\underline{z_r^*}^\alpha - \underline{z_r}^\alpha\right)\Upsilon, \tag{24}$$

where $i = 0$, and

$$\begin{aligned} \frac{\partial \underline{\Psi}^\alpha}{\partial c_i^k} &= \frac{\partial \underline{\Psi}^\alpha}{\partial \underline{z_r}^\alpha}\frac{\partial \underline{z_r}^\alpha}{\partial \underline{c_i}^\alpha}\frac{\partial \underline{c_i}^\alpha}{\partial c_i^k} \\ &= \left(\underline{z_r^*}^\alpha - \underline{z_r}^\alpha\right)\begin{cases} \underline{O_i}^\alpha \Upsilon, & \underline{c_i}^\alpha \ge 0 \\ \overline{O_i}^\alpha \Upsilon, & \underline{c_i}^\alpha < 0, \end{cases} \end{aligned} \tag{25}$$

where $i = 1, \ldots, n$, and

$$\Upsilon = \begin{cases} 1-\alpha & k = 1 \\ \alpha & k = 2 \\ 0 & k = 3, \end{cases} \tag{26}$$

also we have

$$\frac{\partial \overline{\Psi}^\alpha}{\partial c_i^k} = \frac{\partial \overline{\Psi}^\alpha}{\partial \overline{z_r}^\alpha}\frac{\partial \overline{z_r}^\alpha}{\partial \overline{c_i}^\alpha}\frac{\partial \overline{c_i}^\alpha}{\partial c_i^k} = \left(\overline{z_r^*}^\alpha - \overline{z_r}^\alpha\right)\Upsilon_1, \tag{27}$$

where $i = 0$, and

$$\begin{aligned} \frac{\partial \overline{\Psi}^\alpha}{\partial c_i^k} &= \frac{\partial \overline{\Psi}^\alpha}{\partial \overline{z_r}^\alpha}\frac{\partial \overline{z_r}^\alpha}{\partial \overline{c_i}^\alpha}\frac{\partial \overline{c_i}^\alpha}{\partial c_i^k} \\ &= \left(\overline{z_r^*}^\alpha - \overline{z_r}^\alpha\right)\begin{cases} \overline{O_i}^\alpha \Upsilon_1, & \overline{c_i}^\alpha \ge 0 \\ \underline{O_i}^\alpha \Upsilon_1, & \overline{c_i}^\alpha < 0, \end{cases} \end{aligned} \tag{28}$$

where $i = 1, \ldots, n$, and

$$\Upsilon_1 = \begin{cases} 0 & k = 1 \\ \alpha & k = 2 \\ 1-\alpha & k = 3. \end{cases} \tag{29}$$

The coefficient c_i is updated as

$$c_i^k(r+1) = c_i^k(r) - \eta\frac{\partial \Psi_r}{\partial c_i^k}, \tag{30}$$

where $k = 1, 2, 3$ and η is the training rate $\eta > 0$. For the requirement of increasing the training process, the adding of the momentum term is mentioned as

$$c_i^k(r+1) = c_i^k(r) - \eta\frac{\partial \Psi_r}{\partial c_i^k} + \gamma\left[c_i^k(r) - c_i^k(r-1)\right], \tag{31}$$

where $\gamma > 0$.

Learning Algorithm

(1) Step 1: choose the training rates $\eta > 0$, $\gamma > 0$, and the stop criterion $\overline{\Psi} > 0$. The initial fuzzy vector $C = (c_1, \ldots, c_n)$ is selected randomly. The initial learning iteration is $r = 1$, and the initial learning error $\Psi = 0$.

(2) Repeat the following steps for $\alpha = \alpha_1, \ldots, \alpha_m$, until all training data are applied.

   (a) Forward calculation: calculate the α-level of the fuzzy output z_r with the α-level of the fuzzy input w_r, and the fuzzy connection weight C.

   (b) Backpropagation: adjust fuzzy parameters c_i, $i = 1, \ldots, n$, by using the cost function for the α-level of the fuzzy output z_r, and the fuzzy target output z_r^*.

   (c) Stop criterion: calculate the cycle error Ψ_r, $\Psi = \Psi + \Psi_r$. $r = r + 1$. If $\Psi > \overline{\Psi}$, let $\Psi = 0$; a new training cycle is initiated. Go to (a).

4. Upper Bounds of the Modeling Errors

In this section, we extend some well-known approximation theories into fuzzy equation modeling. We first define the modeling error in the sense of fuzzy number.

Definition 5. The distance between two fuzzy numbers, $u, v \in E$, is defined as the Hausdorff metric $d_H(u, v)$:

$$d_H(u, v) = \max\left\{\sup_{x\in u}\inf_{y\in v}|x - y|,\ \sup_{y\in v}\inf_{x\in u}|x - y|\right\}. \tag{32}$$

Lemma 6. *If $\theta \subset E$ is a compact set, then θ is uniformly support-bounded; that is, there is a compact set $U \subset \mathfrak{R}$, such that, $\forall u \in \theta$,*

$$\operatorname{Supp}(u) \subset U. \tag{33}$$

Lemma 7. *Let $u, v \in E$, and $\alpha \in (0, 1]$, $\lambda \in (0, +\infty]$; then one has (i) if $f : \mathfrak{R} \to \mathfrak{R}$ is continuous, $[f(u)]^{\alpha} = f([u^{\alpha}])$ holds; (ii) if $f : \mathfrak{R} \to \mathfrak{R}$ is continuous, then $f(\operatorname{Supp}(u)) = \operatorname{Supp}(f(u))$.*

Proof. We need to only prove (ii) since (i) comes from [39]. At first, we demonstrate $\overline{f(A)} = f(\overline{A})$ for $A \subset \mathfrak{R}$. In fact, since $f(A) \subset f(\overline{A})$, and $f(\overline{A})$ is closed by the continuity of f, hence $\overline{f(A)} \subset f(\overline{A})$. On the other hand, for arbitrarily given $y \in f(\overline{A})$, there is a sequence $\{x_n \mid n \in N\} \subset \mathfrak{R}$ and a $x \in \mathfrak{R}$, such that $x_n \to x$ $(n \to +\infty)$, $y = f(x)$. The continuity of f implies $\lim_{n \to +\infty} f(x_n) = f(x) = y$. But $f(x_n) \in f(A)$, so $y \in \overline{f(A)}$. Hence $f(\overline{A}) \subset \overline{f(A)}$. Thus $\overline{f(A)} = f(\overline{A})$.

Considering

$$\begin{aligned} \operatorname{Supp}(f(u)) &= \overline{\{y \in \mathfrak{R} \mid f(u)(y) > 0\}}, \\ f(\operatorname{Supp}(u)) &= f\left(\overline{\{x \in \mathfrak{R} \mid u(x) > 0\}}\right), \end{aligned} \tag{34}$$

we obtain the fact that

$$\begin{aligned} f(\operatorname{Supp}(u)) &= \overline{f(\{x \in \mathfrak{R} \mid u(x) > 0\})} \\ &= \overline{\{f(x) \in \mathfrak{R} \mid u(x) > 0\}} \end{aligned} \tag{35}$$

holds. Since it may be easily proved that $\{y \in \mathfrak{R} \mid f(u)(y) > 0\} = \{f(x) \mid u(x) > 0\}$, therefore,

$$\operatorname{Supp}(f(u)) = f(\operatorname{Supp}(u)) \tag{36}$$

which implies the lemma. □

Lemma 8. *Let $B \subset \mathfrak{R}$ be a compact set and f, g be continuous on B, $h > 0$; moreover*

$$|f(x) - g(x)| < h, \quad \forall x \in B. \tag{37}$$

Then, for each compact set $B_1 \subset B$, we have $|\sup_{x \in B_1} f(x) - \sup_{x \in B_1} g(x)| < h$.

Proof. Because of the facts that B_1 is a compact set and f, g are continuous on B_1, then there are $x_0 \in B_1$, $y_0 \in B_1$, such that

$$\begin{aligned} f(x_0) &= \sup_{x \in B_1} f(x), \\ g(y_0) &= \sup_{x \in B_1} g(x). \end{aligned} \tag{38}$$

Supposing $|f(x_0) - g(y_0)| \geq h$, we have

$$\begin{aligned} & f(x_0) - g(y_0) \leq -h, \\ \text{or } & f(x_0) - g(y_0) \geq h. \end{aligned} \tag{39}$$

In the first case of (39), because $f(y_0) \leq f(x_0)$,

$$\begin{aligned} & f(y_0) - g(y_0) \leq f(x_0) - g(y_0) \leq -h \Longrightarrow \\ & |f(y_0) - g(y_0)| \geq h \end{aligned} \tag{40}$$

holds, which contradicts (37). In the second case of (39), since $g(x_0) \leq g(y_0)$, we obtain

$$\begin{aligned} & f(x_0) - g(x_0) \geq f(x_0) - g(y_0) \geq h \Longrightarrow \\ & |f(x_0) - g(x_0)| \geq h \end{aligned} \tag{41}$$

which also contradicts (37). Therefore, (39) is not true; hence $-h < f(x_0) - g(y_0) < h$, so $|f(x_0) - g(x_0)| < h$; that is, $|\sup_{x \in B_1} f(x) - \sup_{x \in B_1} g(x)| < h$. The proof is completed. □

Theorem 9. *Let $f : \mathfrak{R} \to \mathfrak{R}$ be a continuous function; then for each compact set $\theta \subset E_0$ (the set of all the bounded fuzzy set) and $\psi > 0$, there are $n \in N$ and $a_0, a_i \in E_0$, $i = 1, 2, \ldots, n$, such that*

$$d\left(f(\tilde{x}), \sum_{i=1}^{n} f_i(x) a_i + a_0\right) < \psi, \quad \forall x \in \theta, \ \forall \tilde{x} \in \mathfrak{R}, \tag{42}$$

where ψ is a finite number.

Proof. The proof of the theorem can be followed from the results below. □

If the function $f : \mathfrak{R} \to \mathfrak{R}$, we can extend f by the extension principle to the fuzzy function which is also written as $f : E_0 \to E$ as follows:

$$f(u)(y) = \bigvee_{f(x) = y} \{u(x)\}, \quad y \in \mathfrak{R}, \ \forall u \in E_0, \tag{43}$$

where f is called the extended function. Moreover, $cc(\mathfrak{R})$ stands for the set of bounded closed intervals of $\mathfrak{R}$. Obviously

$$[u]^{\alpha} \in cc(\mathfrak{R}), \quad u \in E_0 \Longrightarrow \forall \alpha \in (0, 1]. \tag{44}$$

Moreover

$$\operatorname{Supp}(u) \in cc(\mathfrak{R}). \tag{45}$$

So from now on, we suppose

$$\operatorname{Supp}(u) = [s_1(u), s_2(u)]. \tag{46}$$

Theorem 10. *Let $f : \mathfrak{R} \to \mathfrak{R}$ be a continuous function; then for each compact set $\theta \subset E_0$, $\varrho > 0$, and arbitrary $\varepsilon > 0$, there are $n \in N$ and $a_0, a_i \in E_0$, $i = 1, 2, \ldots, n$, such that*

$$d\left(f(x), \sum_{i=1}^{n} f_i(x) a_i + a_0\right) < \varrho, \quad \forall x \in \theta, \tag{47}$$

where ϱ is a finite number. The lower and the upper limits of the α-level set of fuzzy function diminish to ϱ, but the center goes to ε.

Proof. Because $\theta \subset E_0$ is a compact set, hence, by Lemma 6, we consider $U \subset \mathfrak{R}$ to be the compact set corresponding to θ. $\forall \varepsilon > 0$, by the conclusions in [27], there are $n \in N$ and $a_0, a_i \in \mathfrak{R}$, $i = 1, 2, \ldots, n$, such that

$$\left| f(x) - \sum_{i=1}^{n} f_i(x) a_i + a_0 \right| < \varepsilon, \quad \forall x \in U, \tag{48}$$

holds. Let $g(x) = \sum_{i=1}^{n} f_i(x) a_i + a_0$, $x \in \mathfrak{R}$; then

$$\left| f(x) - g(x) \right| < \varepsilon, \quad \forall x \in U. \tag{49}$$

By Theorem 11, we imply that (47) holds. □

Theorem 11. *Supposing $\theta \subset E_0$ is compact, U the corresponding compact set of θ and $f, g : \mathfrak{R} \to \mathfrak{R}$ are the continuous functions which satisfy the condition that, for given $h > 0$,*

$$\left| f(x) - g(x) \right| < h, \quad \forall x \in U, \tag{50}$$

holds. Then, $\forall u \in \theta$, $d(f(u) - g(u)) \leq h$.

Proof. Let $u \in E$ and $\alpha \in (0, 1]$. Because f, g are continuous, hence $[f(u)]^\alpha = f([u^\alpha])$, $[g(u)]^\alpha = g([u^\alpha])$ holds by Lemma 7. Therefore, we obtain the following facts by the conclusions from [40]:

$$\begin{aligned} d_H\left([f(u)]^\alpha - [g(u)]^\alpha\right) &= d_H\left(f([u^\alpha]) - g([u^\alpha])\right) \\ &= \sup_{|p|=1} \left\{ \left| s(p, f([u^\alpha])) - s(p, g([u^\alpha])) \right| \right\}. \end{aligned} \tag{51}$$

Because, for $p \in \mathfrak{R}$: $|p| = 1$, we have that

$$\begin{aligned} &\left| sp, f([u^\alpha]) - s(p, g([u^\alpha])) \right| \\ &\quad = \left| \sup \{ py \mid y \in f([u^\alpha]) \} \right. \\ &\quad \left. - \sup \{ py \mid y \in g([u^\alpha]) \} \right| \\ &\quad = \left| \sup \{ pf(x) \mid x \in [u]^\alpha \} \right. \\ &\quad \left. - \sup \{ pg(x) \mid x \in [u]^\alpha \} \right| \end{aligned} \tag{52}$$

holds. And considering the conditions in the theorem, we obtain

$$\left| pf(x) - pg(x) \right| = \left| f(x) - g(x) \right| < h, \quad \forall x \in [u]^\alpha. \tag{53}$$

Therefore, by (51), (52), and Lemma 8, the following

$$\begin{aligned} &d_H\left([f(u)]^\alpha, [g(u)]^\alpha\right) < h \Longrightarrow \\ &d(f(u), g(u)) = \sup_{\alpha \in (0,1]} \left\{ d_H\left([f(u)]^\alpha, [g(u)]^\alpha\right) \right\} \\ &\quad \leq h, \quad \forall \alpha \in (0, 1], \end{aligned} \tag{54}$$

holds, which proves the theorem. □

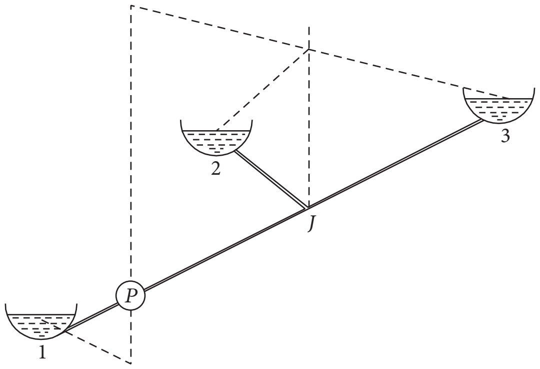


FIGURE 2: Pumping water from one tank to the other two tanks [37].

5. Numerical Examples

In this section, we use four examples to show how to apply fuzzy equations to model uncertain nonlinear systems. Here the learning rate in (30) is $\eta = 0.001$; the momentum constant in (31) is $\gamma = 0.001$.

Example 1. There are three tanks connected to a pipeline at a constant H; see Figure 2. We want to pump water from one tank to the other two tanks. This system satisfies the following fuzzy relation

$$H = A_0 \oplus A_1 F_1 \oplus A_2 F_2 \oplus A_3 F_3, \tag{55}$$

where $F_1 = \sqrt{2x}$, $F_2 = x\sqrt{x}$, and $F_3 = x^3$ are the quantity of flow and x is the elapsed time. H is the high of the pipe, A_0, A_1, A_2, and A_3 are the characteristic coefficients of the pump; they satisfy triangular uncertainty (5):

$$\begin{aligned} A_0 &= (1, 5, 8), \\ A_1 &= (3, 7, 8), \\ A_2 &= (1, 2, 4), \\ A_3 &= (1, 3, 4). \end{aligned} \tag{56}$$

We use four types of input to train the neural network, whose weights are the fuzzy parameters of (55). The input data are

$$x = \{2, (2, 4, 5), (3, 5, 6, 7), (1, 2, 4)\}, \tag{57}$$

where $(2, 4, 5)$ and $(1, 2, 4)$ satisfy the triangle function (5), $(3, 5, 6, 7)$ is the trapezoidal function (6), and 2 is a crisp number. These inputs are applied to (55); the corresponding outputs data are

$$\begin{aligned} H = \{&(17.82, 48.65, 67.31), (17.82, 224.79, 544.47), \\ &(40.54, 413.31, 691.94, 1344.45), \\ &(7.24, 48.65, 318.62)\}. \end{aligned} \tag{58}$$

Now these input/output pairs are used repeatedly to train the neural network. The weights are A_0, A_1, A_2, and A_3. The training results are shown in Table 1. We can see that the fuzzy characteristic coefficients converge to their real values.

Table 1: Neural network approximation for the coefficients.

t	A_0	A_1	A_2	A_3
1	$(3.9, 7.9, 10.9)$	$(5.9, 9.8, 10.9)$	$(2.91, 3.9, 5.9)$	$(3.9, 5.9, 6.9)$
2	$(3.7, 7.7, 10.6)$	$(5.7, 9.6, 10.7)$	$(2.7, 3.71, 5.8)$	$(3.7, 5.7, 6.8)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
75	$(1.0, 5.0, 8.0)$	$(3.0, 7.0, 8.0)$	$(1.0, 2.0, 4.0)$	$(1.0, 3.0, 4.0)$

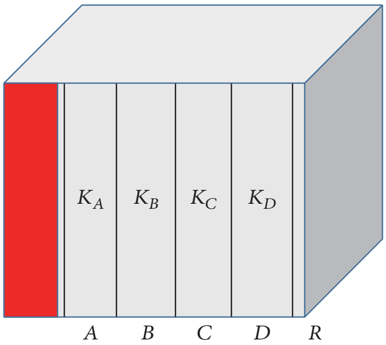


Figure 3: Heat source by insulating materials [37].

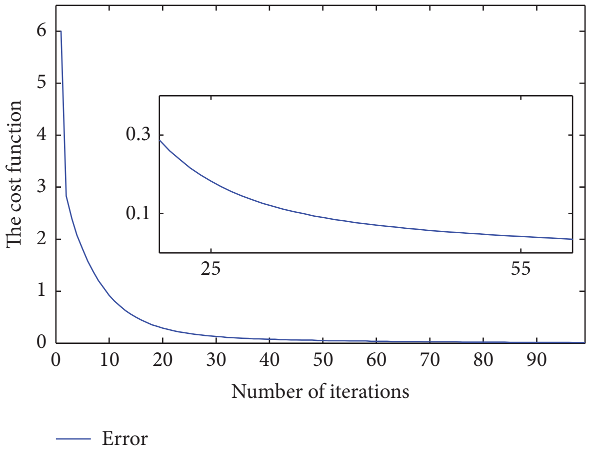


Figure 4: Neural network approximation errors.

Example 2. The heat source is on the left of the insulating materials; see Figure 3. The conductivity coefficients of these materials are $1/K_A = e^x$, $1/K_B = x\sqrt{2x}$, and $1/K_C = \sqrt{x}$; here x is the elapsed time. The thermal balance is as follows [41]:

$$R = \frac{A}{K_A} \oplus \frac{B}{K_B} \oplus \frac{C}{K_C}. \tag{59}$$

The real parameters are

$$\begin{aligned} A &= (3, 4, 6), \\ B &= (1, 4, 5), \\ C &= (2, 3, 4). \end{aligned} \tag{60}$$

We have the following three types of input:

$$x = \{(2, 3, 4), (3, 5, 6), (1, 3, 5, 7)\}, \tag{61}$$

where $(2, 3, 4)$ and $(3, 5, 6)$ satisfy the triangle function (5) and $(1, 3, 5, 7)$ is the trapezoidal function (6). The corresponding outputs data are

$$\begin{aligned} R = \{&(28.99, 114.93, 392.15), (71.06, 663.6, 2534.29), \\ &(11.56, 95.96, 616.17, 6616.37)\}. \end{aligned} \tag{62}$$

We repeat them for neural network training. The approximation errors are shown in Figure 4. After 99 times, the parameters A, B, and C converge.

Example 3. The water in the pipe d_1 is divided into three pipes d_2, d_3, and d_4; see Figure 5. The areas of the pipes are uncertain; they satisfy the triangle function (5). $A_2 = (1, 3, 5)$, $A_3 = (3, 5, 6)$, and $A_4 = (2, 4, 5)$. The water velocities in the pipes are controlled by the valves parameters x, $v_2 = e^x$, $v_3 = x$, and $v_4 = \sqrt{x}$ [42]. The discharge of the water cross in pipe d_1 is

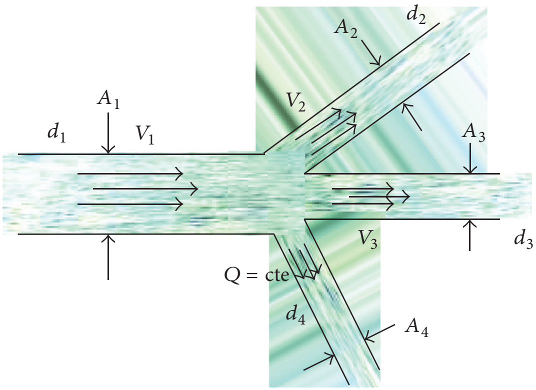


Figure 5: Water channel system.

$$Q = A_2 v_2 \oplus A_3 v_3 \oplus A_4 v_4. \tag{63}$$

We have the following three types of input:

$$x = \{(1, 2, 3), (2, 3, 5), (3, 4, 6)\}, \tag{64}$$

where $(1, 2, 3)$, $(2, 3, 5)$, and $(3, 4, 6)$ satisfy the triangle function (5). The corresponding outputs data are

$$\begin{aligned} Q = \{&(7.71, 59.99, 327.94), (23.6, 202.69, 3751.51), \\ &(72.72, 683.17, 12151.11)\}. \end{aligned} \tag{65}$$

Table 2: Neural network approximation for the coefficients.

t	A_2	A_3	A_4
1	(3.9, 5.9, 7.9)	(5.8, 7.9, 8.9)	(4.9, 6.8, 7.8)
2	(3.7, 5.6, 7.7)	(5.7, 7.7, 8.7)	(4.7, 6.6, 7.7)
3	(3.5, 5.4, 7.5)	(5.6, 7.5, 8.6)	(4.6, 6.4, 7.5)
⋮	⋮	⋮	⋮
57	(1.0, 3.0, 5.0)	(3.0, 5.0, 6.0)	(2.0, 4.0, 5.0)

Table 3: Neural network approximation for the coefficients.

t	$-L_1/(A * EE)$	$-L_2/(A * EE)$	$-L_3/(A * EE)$
1	(−8.8, −6.9, −5.8)	(−9.9, −8.9, −5.91)	(−6.8, −5.7, −4.9)
2	(−8.7, −6.7, −5.7)	(−9.7, −8.7, −5.8)	(−6.6, −5.6, −4.8)
3	(−8.5, −6.6, −5.5)	(−9.6, −8.6, −5.6)	(−6.5, −5.5, −4.7)
⋮	⋮	⋮	⋮
69	(−5.0, −3.0, −2.0)	(−6.0, −5.0, −2.0)	(−3.0, −2.0, −1.0)

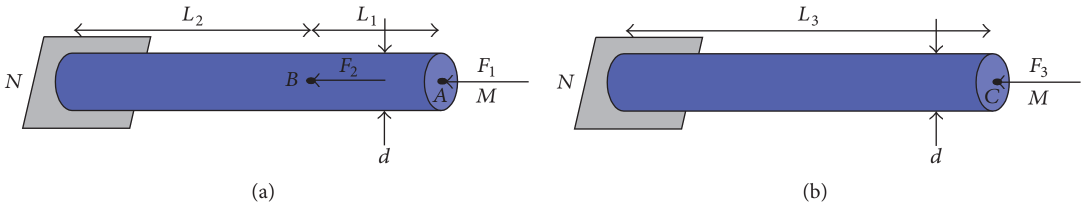


Figure 6: Two solid cylindrical rods.

We repeat the above input/output 57 times. The neural approximation results of A_2, A_3, and A_4 are shown in Table 2.

Example 4. The deformation of a solid cylindrical rod made of special foam depends on stiffness E, three different forces on it F_1, F_2, and F_3, the positions of the forces L_1, L_2, and L_3, and the diameter of the rod d [36]; see Figure 6. The area of the rod is $A = (\pi/4)d^2$. The external forces are the function of , $F_1 = x^2$, $F_2 = \sqrt{x}$, and $F_3 = 2x$ [43]. According to the tension relations [43],

$$\frac{-L_1(F_1+F_2+F_3)}{A * EE} \oplus \frac{-L_2(F_2+F_3)}{A * EE} \oplus \frac{-L_3F_3}{A * EE} = N, \quad (66)$$

where $d = 0.02$, $EE = 2 \times 10^3$, and

$$\frac{-L_1}{AE} = (-5, -3, -2),$$
$$\frac{-L_2}{AE} = (-6, -5, -2), \quad (67)$$
$$\frac{-L_3}{AE} = (-3, -2, -1).$$

We have the following three types of input:

$$x = \{(1, 3, 4), (5, 6, 7), (2, 3, 5)\}, \quad (68)$$

where (1, 3, 4), (5, 6, 7), and (2, 3, 5) satisfy the triangle function (5). The corresponding outputs data are

$$N = \{(-77, -36, -5), (-194, -105, -37), (-110, -36, -10)\}. \quad (69)$$

We use a neural network to approximate $-L_1/AE$, $-L_2/AE$, and $-L_3/AE$. Convergence is reached after 69 times training. The results are shown in Table 3. To compare our method with the other numerical methods, we use the fuzzy cubic spline method [44] to approximate the solutions of the fuzzy equation (66). The comparison results are shown in Figure 7. We can see that both the neural networks based algorithm (our method) and the fuzzy cubic spline method can approximate the solutions of the fuzzy equations. The approximation errors of the neural networks based algorithm are much smaller than the fuzzy cubic spline method. Also the convergence speed of the neural networks based algorithm is faster. The fuzzy cubic spline method is not robust at the beginning stage.

6. Conclusion

In this paper, the uncertainties in nonlinear systems are modeled by the fuzzy equations and fuzzy numbers. However, the parameters of the fuzzy numbers and fuzzy equations cannot be obtained directly. We construct neural models with the similar structure as the fuzzy equations. By modifying the

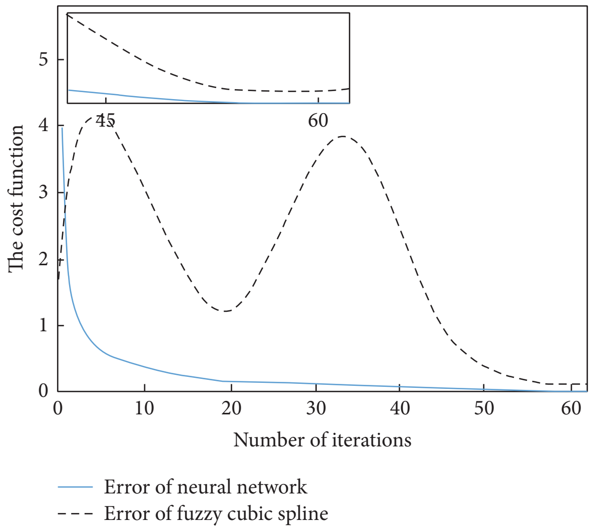


Figure 7: Approximation errors of neural network and fuzzy cubic spline.

backpropagation method, the neural networks are trained. The coefficients of the fuzzy equations are approximated by the neural networks. The upper bounds of the fuzzy modeling errors are proven. We successfully extend the approximation theory of crisp models to fuzzy equation model. The novel modeling method is validated with four benchmark examples.

Competing Interests

The authors declare that they have no competing interests.

References

[1] L. A. Zadeh, "Toward a generalized theory of uncertainty (GTU)—an outline," *Information Sciences. An International Journal*, vol. 172, no. 1-2, pp. 1–40, 2005.

[2] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 15, no. 1, pp. 116–132, 1985.

[3] E. H. Mamdani, "Application of fuzzy algorithms for control of simple dynamic plant," *Proceedings of the Institution of Electrical Engineers*, vol. 121, no. 12, pp. 1585–1588, 1974.

[4] D. Boffi and L. Gastaldi, "Interpolation estimates for edge finite elements and application to band gap computation," *Applied Numerical Mathematics*, vol. 56, no. 10-11, pp. 1283–1292, 2006.

[5] M. Mastyło, "Interpolation estimates for entropy numbers with applications to non-convex bodies," *Journal of Approximation Theory*, vol. 162, no. 1, pp. 10–23, 2010.

[6] R. D. Neidinger, "Multivariable interpolating polynomials in newton forms," in *Proceedings of the Joint Mathematics Meetings*, pp. 5–8, Washington, DC, USA, 2009.

[7] V. Barthelmann, E. Novak, and K. Ritter, "High dimensional polynomial interpolation on sparse grids," *Advances in Computational Mathematics*, vol. 12, no. 4, pp. 273–288, 2000.

[8] H. Schroder, V. K. Murthy, and E. V. Krishnamurthy, "Systolic algorithm for polynomial interpolation and related problems," *Parallel Computing*, vol. 17, no. 4-5, pp. 493–503, 1991.

[9] A. Zolic, *Numerical Mathematics*, Faculty of Mathematics, Belgrade, Serbia, 2008.

[10] P. J. Olver, "On multivariate interpolation," *Studies in Applied Mathematics*, vol. 116, no. 2, pp. 201–240, 2006.

[11] J. Szabados and P. Vértesi, *Interpolation of functions*, World Scientific Publishing Co., Singapore, 1990.

[12] V. M. Tikhomirov, "Approximation theory," in *Analysis II*, R. V. Gamkrelidze, Ed., vol. 14 of *Encyclopaedia of Mathematical Sciences*, Springer, Berlin, Germany, 1990.

[13] D. Xiu and J. S. Hesthaven, "High-order collocation methods for differential equations with random inputs," *SIAM Journal on Scientific Computing*, vol. 27, no. 3, pp. 1118–1139, 2005.

[14] M. Friedman, M. Ming, and A. Kandel, "Fuzzy linear systems," *Fuzzy Sets and Systems*, vol. 96, no. 2, pp. 201–209, 1998.

[15] J. J. Buckley and Y. Qu, "Solving linear and quadratic fuzzy equations," *Fuzzy Sets and Systems*, vol. 38, no. 1, pp. 43–59, 1990.

[16] S. Abbasbandy, "The application of homotopy analysis method to nonlinear equations arising in heat transfer," *Physics Letters A*, vol. 360, no. 1, pp. 109–113, 2006.

[17] S. Abbasbandy and R. Ezzati, "Newton's method for solving a system of fuzzy nonlinear equations," *Applied Mathematics and Computation*, vol. 175, no. 2, pp. 1189–1199, 2006.

[18] T. Allahviranloo, M. Otadi, and M. Mosleh, "Iterative method for fuzzy equations," *Soft Computing*, vol. 12, no. 10, pp. 935–939, 2008.

[19] R. Goetschel and W. Voxman, "Elementary calculus," *Fuzzy Sets and Systems*, vol. 18, no. 1, pp. 31–43, 1986.

[20] M. Mazandarani and A. V. Kamyad, "Modified fractional Euler method for solving fuzzy fractional initial value problem," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 1, pp. 12–21, 2013.

[21] S. Salahshour, T. Allahviranloo, and S. Abbasbandy, "Solving fuzzy fractional differential equations by fuzzy Laplace transforms," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 3, pp. 1372–1381, 2012.

[22] H. Wang and Y. Liu, "Existence results for fuzzy integral equations of fractional order," *International Journal of Mathematical Analysis*, vol. 5, no. 17-20, pp. 811–818, 2011.

[23] B. Llanas and F. J. Sainz, "Constructive approximate interpolation by neural networks," *Journal of Computational and Applied Mathematics*, vol. 188, no. 2, pp. 283–308, 2006.

[24] M. Y. Waziri and Z. A. Majid, "A new approach for solving dual fuzzy nonlinear equations using Broyden's and Newton's methods," *Advances in Fuzzy Systems*, vol. 2012, Article ID 682087, 5 pages, 2012.

[25] S. Pederson and M. Sambandham, "The Runge-Kutta method for hybrid fuzzy differential equations," *Nonlinear Analysis. Hybrid Systems*, vol. 2, no. 2, pp. 626–634, 2008.

[26] M. Kajani and B. V. Lupulescu, "On a class of fuzzy functional differential equations," *Fuzzy Sets and Systems*, vol. 160, no. 11, pp. 1547–1562, 2009.

[27] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems*, vol. 2, no. 4, pp. 303–314, 1989.

[28] W. Yu and X. Li, "Fuzzy identification using fuzzy neural networks with stable learning algorithms," *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 3, pp. 411–420, 2004.

[29] J. J. Buckley and E. Eslami, "Neural net solutions to fuzzy problems: the quadratic equation," *Fuzzy Sets and Systems*, vol. 86, no. 3, pp. 289–298, 1997.

[30] A. Jafarian, R. Jafari, A. K. Golmankhaneh, and D. Baleanu, "Solving fully fuzzy polynomials using feed-back neural networks," *International Journal of Computer Mathematics*, vol. 92, no. 4, pp. 742–755, 2015.

[31] Jafarian and M. S. A. Measoomy Nia, "Solving fuzzy polynomials using neural nets with a new learning algorithm," *Australian Journal of Basic and Applied Sciences*, vol. 5, no. 9, pp. 2295–2301, 2011.

[32] R. Jafari and W. Yu, "Fuzzy control for uncertainty nonlinear systems with dual fuzzy equations," *Journal of Intelligent & Fuzzy Systems*, vol. 29, no. 3, pp. 1229–1240, 2015.

[33] A. Jafarian and R. Jafari, "Approximate solutions of dual fuzzy polynomials by feed-back neural networks," *Journal of Soft Computing and Applications*, vol. 2012, Article ID jsca-00005, 16 pages, 2012.

[34] M. Mosleh, "Evaluation of fully fuzzy matrix equations by fuzzy neural network," *Applied Mathematical Modelling*, vol. 37, no. 9, pp. 6364–6376, 2013.

[35] Y. Ito, "Independence of unscaled basis functions and finite mappings by neural networks," *The Mathematical Scientist*, vol. 26, no. 2, pp. 117–126, 2001.

[36] A. R. Tahavvor and M. Yaghoubi, "Analysis of natural convection from a column of cold horizontal cylinders using Artificial Neural Network," *Applied Mathematical Modelling*, vol. 36, no. 7, pp. 3176–3188, 2012.

[37] R. Jafari and W. Yu, "Uncertainty nonlinear systems modeling with fuzzy equations," in *Proceedings of the 16th IEEE International Conference on Information Reuse and Integration (IRI '15)*, pp. 182–188, San Francisco, Calif, USA, August 2015.

[38] J. A. K. Suykens, J. De Brabanter, L. Lukas, and J. Vandewalle, "Weighted least squares support vector machines: robustness and sparse approximation," *Neurocomputing*, vol. 48, no. 1–4, pp. 85–105, 2002.

[39] L. B. Yang and Y. Y. Gao, *Fuzzy Mathematics-Theory and its Application*, South China University of Technology, Guangzhou, China, 1993.

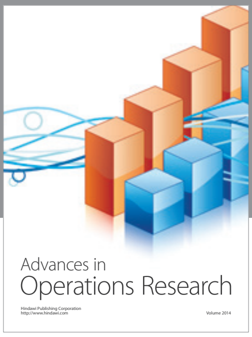
[40] H. Rådström, "An embedding theorem for spaces of convex sets," *Proceedings of the American Mathematical Society*, vol. 3, pp. 165–169, 1952.

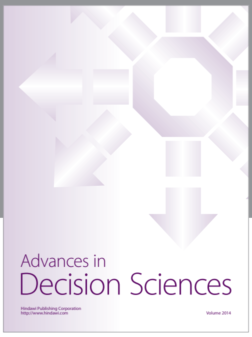
[41] J. P. Holman, *Heat Transfer*, McGraw-Hill, New York, NY, USA, 1997.

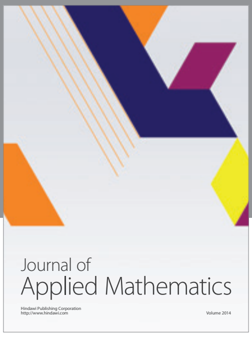
[42] V. L. Streeter, E. B. Wylie, and E. Benjamin, *Fluid Mechanics*, Mc.Graw-Hill Book Company, 1999.

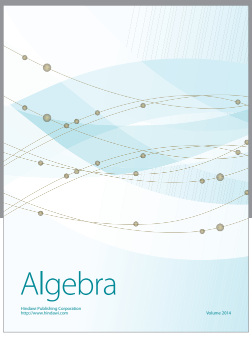
[43] F. P. Beer and E. R. Johnston, *Mechanics of Materials*, McGraw-Hill, 2nd edition, 1992.

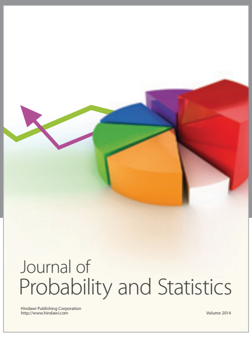
[44] M. C. Floreno and G. Novelli, "Implementing fuzzy polynomial interpolation (FPI) and fuzzy linear regression (LFR)," *Le Matematiche*, vol. 51, no. 1, pp. 59–76, 1996.


Advances in
Operations Research

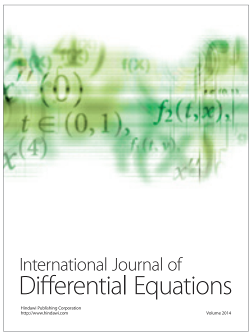

Advances in
Decision Sciences


Journal of
Applied Mathematics

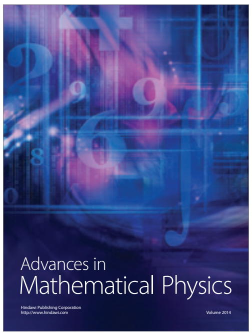

Algebra

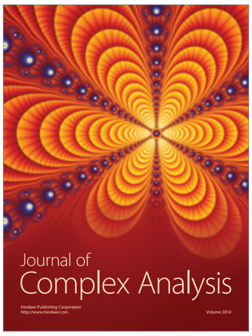

Journal of
Probability and Statistics

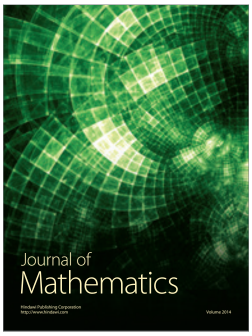
The Scientific
World Journal

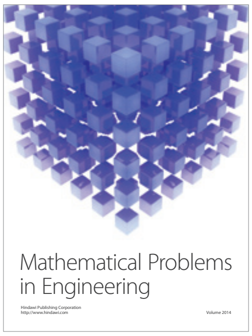

International Journal of
Differential Equations

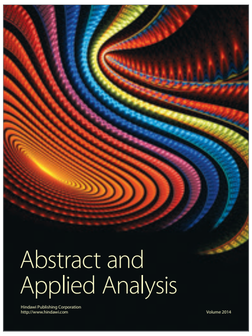
International Journal of
Combinatorics

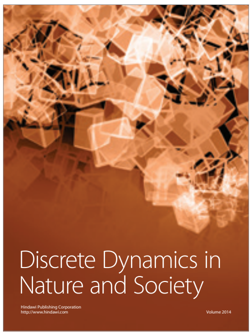

Advances in
Mathematical Physics

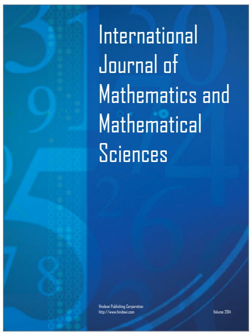

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

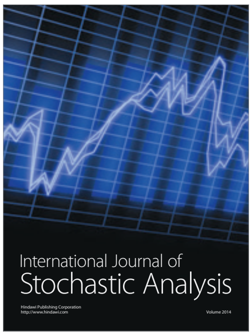

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization