ELSEVIER

Fuzzy Sets and Systems 130 (2002) 175–188

FUZZY sets and systems

www.elsevier.com/locate/fss

Universal approximation by hierarchical fuzzy system with constraints on the fuzzy rule

Moon G. Joo[a], Jin S. Lee[b,*]

[a] Research Institute of Industrial Science and Technology, Pohang 790-330, South Korea
[b] Department of Electrical Engineering, Pohang University of Science and Technology, Pohang 790-784, South Korea

Received 18 July 2000; received in revised form 10 August 2001; accepted 22 August 2001

Abstract

This paper presents a special hierarchical fuzzy system where the outputs of the previous layer are not used in the IF-parts, but used only in the THEN-parts of the fuzzy rules of the current layer.

The proposed scheme can be shown to be a universal approximator to any continuous function on a compact set if complete fuzzy sets are used in the IF-parts of the fuzzy rules with singleton fuzzifier and center average defuzzifier.

From the simulation of ball and beam control system, it is demonstrated that the proposed scheme approximates with good accuracy the model nonlinear controller with fewer fuzzy rules than the centralized fuzzy system and its control performance is comparable to that of the nonlinear controller.

Keywords: Hierarchical fuzzy logic system; Universal approximation; Stone–Weierstrass theorem; Fuzzy control

1. Introduction

One of the important issues in fuzzy logic systems is how to reduce the number of involved fuzzy rules and their corresponding computation requirements. In fact, the number of fuzzy rules grows exponentially with the number of input variables. Specifically, a single-output fuzzy logic system with n input variables and m membership functions defined for each input variable requires m^n number of fuzzy rules. To overcome the problem, the idea of using hierarchical structure in designing a fuzzy system has been reported in 1991 by Raju and Zhou [16,17], where input variables are put into a collection of low-dimensional fuzzy logic units (FLUs) and the outputs of the FLUs are used as input variables for the FLU at the next layer. According to them, the number of fuzzy rules that are employed in the hierarchical fuzzy system (HFS) is shown to be proportional to the number of input variables.

In HFS, however, it is not trivial to retrieve physical meanings from the outputs of the FLUs at the previous layer with or without supplementary input variables. Consequently, if they are used as input variables for the

* Corresponding author. Tel.: +82-542792230; fax: +82-542792903.
E-mail address: jsoo@postech.ac.kr (J.S. Lee).

0165-0114/02/$ - see front matter
PII: S0165-0114(01)00176-2

FLUs at the next layer, as is usually the case in HFS, then the involved fuzzy rules in the middle of a hierarchical structure have little physical meaning, and consequently, are difficult to design. This phenomenon becomes prominent as the number of layers grows in HFS. To overcome the problem, we propose an HFS where the outputs at the previous layer are never used in the IF-parts, but used only in the THEN-parts of fuzzy rules at the next layer. As a result, all of the IF-parts of fuzzy rules use only the original input variables, and thus, the resulting fuzzy rules come with clear physical meaning and become easy to design.

On the other hand, Wang proved in 1998 that a certain class of HFS can serve as a universal approximator to any continuous function on a compact set. In his scheme [14,15,2], the ith rule of the FLU at the kth layer is given by

$$\text{IF } x_{N(k-1)+1} \text{ is } A^i_{N(k-1)+1} \text{ and} \ldots \text{and } x_{N(k)} \text{ is } A^i_{N(k)} \text{ and } y_{k-1} \text{ is } B^i_k$$

$$\text{THEN } \bar{y}^i_k \text{ is } \sum_{j=0}^{m^{N(k-1)-1}-1} q^i_{kj}(y_{k-1})^j, \tag{1}$$

where $N(k)=\sum_{l=1}^{k} n_l$, n_k is the number of involved input variables at the kth layer, $x_{N(k-1)+1},\ldots,x_{N(k)}$ are the input variables, y_{k-1} is the output of the FLU at the $(k-1)$th layer, $A^i_{N(k-1)+1},\ldots,A^i_{N(k)}$ and B^i_k are the fuzzy sets, and q^i_{kj}'s are the coefficients.

Huwendiek and Brockmann [6–8] proved that network of fuzzy adaptive nodes (NetFAN), a special class of HFS, can be used as a universal approximator to any continuous function on a compact set. NetFAN uses trapezoidal membership functions, product inference engine, singleton fuzzifier, and center average defuzzifier and the universal approximation capability is proved by using the Stone–Weierstrass theorem. In their scheme, the ith rule of the FLU at the kth layer is given by

$$\text{IF } x_{k1} \text{ is } A^i_{k1} \text{ and } x_{k2} \text{ is } A^i_{k2} \text{ and} \ldots \text{and } y_{(k-1)1} \text{ is } B^i_{k1} \text{ and } y_{(k-1)2} \text{ is } B^i_{k2} \text{ and} \ldots$$

$$\text{THEN } \bar{y}^i_k \text{ is } r^i_k, \tag{2}$$

where x_{kj}'s and $y_{(k-1)j}$'s can be used in other FLU's, A^i_{kj}'s and B^i_{kj}'s are fuzzy sets, and r^i_k's are constants.

As with [8], we prove that the proposed HFS can also serve as a universal approximator to any continuous function on a compact set. However, in contrast with [8], the theorem holds for many types of membership functions as long as complete fuzzy sets are used for input variables with singleton fuzzifier and center average defuzzifier.

This paper is organized as follows: Section 2 introduces the proposed HFS and Section 3 presents the universal approximation capability of the proposed HFS. Section 4 shows the simulation of the control problem of ball and beam system to demonstrate the feasibility of the proposed scheme. Finally, Section 5 summarizes results and draws conclusions.

2. Proposed HFS

The proposed HFS with λ ($\geqslant 2$) layer hierarchy is of the structure as shown in Fig. 1, where $\mathbf{u}$ is a collection of input variables for HFS, $\mathbf{u}=(u_1,u_2,\ldots,u_n)^{\mathrm{T}}\in\mathbb{R}^n$, $d_1,d_2,\ldots,d_\lambda$ are the number of FLUs at each layer, $\mathbf{x}_{k1},\mathbf{x}_{k2},\ldots,\mathbf{x}_{kd_k}$ are the involved input variables for each FLU at the kth layer, $y_{k1},y_{k2},\ldots,y_{kd_k}$ are the outputs of the FLUs at the kth layer, and $y_{\lambda 1}$ is the output of the HFS. Without loss of generality, y_0's are set to 0 and the HFS is to be a feed forward network.

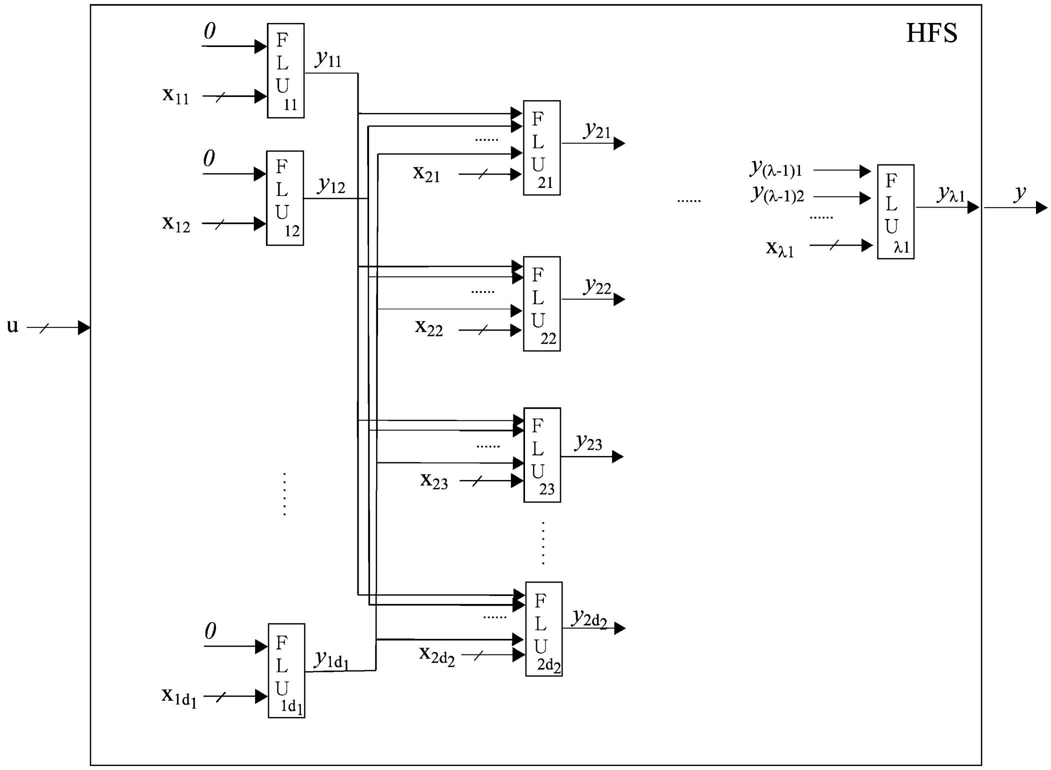


Fig. 1. The proposed HFS with λ layer hierarchy.

For the dth FLU ($1 \leqslant d \leqslant d_k$) at the kth layer ($1 \leqslant k \leqslant \lambda$) with $n(kd)$ number of input variables, its ith fuzzy rule is written as follows:

$$\text{IF } x_1 \text{ is } A_1^i \text{ and } x_2 \text{ is } A_2^i \text{ and} \ldots \text{and } x_{n(kd)} \text{ is } A_{n(kd)}^i$$

$$\text{THEN } \bar{y}_{kd}^i \text{ is } \sum_{j=1}^{N(kd)} p_j^i \varrho_j + \sum_{(u,v)\in {}_{N(kd)}C_2} w_{uv}^i \varrho_u \varrho_v + r^i, \tag{3}$$

where $N(kd) = n(kd) + d_{k-1}$, $\mathbf{x}_{kd} = (x_1,\ x_2, \ldots, x_{n(kd)})^{\mathrm{T}} \subset \mathbf{u}$, $A_1^i,\ A_2^i, \ldots, A_{n(kd)}^i$ are the input fuzzy sets, p_j^i, w_{uv}^i, r^i are the coefficients in the THEN-part, ϱ_j is defined as

$$\varrho_j = \begin{cases} x_j & \text{if } (1 \leqslant j \leqslant n(kd)), \\ y_{(k-1)(j-n(kd))} & \text{if } (n(kd)+1 \leqslant j \leqslant N(kd)) \end{cases} \tag{4}$$

and ${}_{N(kd)}C_2$ denotes the number of combinations that selects two elements out of $N(kd)$ elements. Note that y_{k-1} is never used in the IF-parts of fuzzy rules because it is just the output of the previous layer with less or no physical meaning.

Now let us define

$$\alpha_{uv} = \frac{\sum_i \mu^i(\mathbf{x}_{kd}) w^i_{uv}}{\sum_i \mu^i(\mathbf{x}_{kd})},$$

$$\beta_j = \frac{\sum_i \mu^i(\mathbf{x}_{kd}) p^i_j}{\sum_i \mu^i(\mathbf{x}_{kd})},$$

$$\gamma = \frac{\sum_i \mu^i(\mathbf{x}_{kd}) r^i}{\sum_i \mu^i(\mathbf{x}_{kd})}, \tag{5}$$

where $\mu^i(\mathbf{x}_{kd})$ is the firing strength of the ith rule given by

$$\mu^i(\mathbf{x}_{kd}) = \begin{cases} \prod_{j=1}^{n(kd)} \mu^i_{A_j}(x_j) & \text{if prod-operation,} \\ \min(\mu^i_{A_1}(x_1), \mu^i_{A_2}(x_2), \ldots, \mu^i_{A_{n(kd)}}(x_{n(kd)})) & \text{if min-operation.} \end{cases}$$

Since $\alpha, \beta, \gamma, w, p, r$ are defined for the dth FLU at the kth layer, they should have k and d as subscripts. In order to improve readability, however, k and d are completely omitted in their notations.

Using singleton fuzzifier, center average defuzzifier, and definition (3)–(5), we have

$$\begin{aligned} y_{kd} &= \frac{\sum_i \mu^i(\mathbf{x}_{kd}) \bar{y}^i_{kd}}{\sum_i \mu^i(\mathbf{x}_{kd})} \\ &= \frac{\sum_i \mu^i(\mathbf{x}_{kd})(\sum_{j=1}^{N(kd)} p^i_j \varrho_j + \sum_{(u,v)} w^i_{uv} \varrho_u \varrho_v + r^i)}{\sum_i \mu^i(\mathbf{x}_{kd})} \\ &= \frac{\sum_i \mu^i(\mathbf{x}_{kd}) \sum_{j=1}^{N(kd)} p^i_j \varrho_j}{\sum_i \mu^i(\mathbf{x}_{kd})} + \frac{\sum_i \mu^i(\mathbf{x}_{kd}) \sum_{(u,v)} w^i_{uv} \varrho_u \varrho_v}{\sum_i \mu^i(\mathbf{x}_{kd})} + \frac{\sum_i \mu^i(\mathbf{x}_{kd}) r^i}{\sum_i \mu^i(\mathbf{x}_{kd})} \\ &= \sum_{j=1}^{N(kd)} \varrho_j \frac{\sum_i \mu^i(\mathbf{x}_{kd}) p^i_j}{\sum_i \mu^i(\mathbf{x}_{kd})} + \sum_{(u,v)} \varrho_u \varrho_v \frac{\sum_i \mu^i(\mathbf{x}_{kd}) w^i_{uv}}{\sum_i \mu^i(\mathbf{x}_{kd})} + \frac{\sum_i \mu^i(\mathbf{x}_{kd}) r^i}{\sum_i \mu^i(\mathbf{x}_{kd})} \\ &= \sum_{j=1}^{N(kd)} \beta_j \varrho_j + \sum_{(u,v) \in {}_{N(kd)}C_2} \alpha_{uv} \varrho_u \varrho_v + \gamma. \end{aligned} \tag{6}$$

In (6), y_{kd} is composed of additions and multiplications of involved variables, which are multiplied by the contribution of input variables defined by β and α.

Remark 1. Note that Wang's HFS uses each input variable only once for some FLU but Huwendiek's HFS uses the same input variable in many FLUs. Both schemes have their own drawbacks. The former has less number of fuzzy rules than the centralized fuzzy system, but the THEN-parts of the fuzzy rules are complicated and in the given kth layer ($k = 1, \ldots, \lambda - 1$), every consequent variables $\bar{y}^{i}_k$'s should be of different values, i.e., $\bar{y}^i_k \neq \bar{y}^j_k$, if $i \neq j$, which may not be easily satisfied in the real case. In the latter scheme, however, the THEN-parts of fuzzy rules are simple, but it may have larger number of fuzzy rules than the centralized

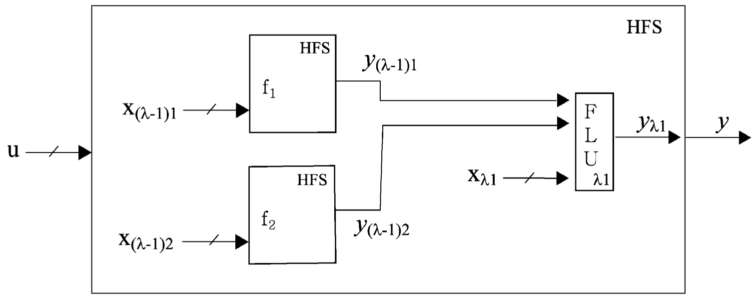


Fig. 2. HFS for Lemmas 1 and 2.

fuzzy system because the input variables used in one FLU maybe used repeatedly in other FLUs. In many applications, however, the number of fuzzy rules are less than that of the centralized fuzzy system as shown in [8] and in the example of this paper. The structure of the latter scheme is adopted in this paper.

3. Universal approximation

In this section, we prove that the proposed HFS can serve as a universal approximator to any continuous function on a compact set. This theorem holds for a number of membership functions such as triangular functions, trapezoidal functions, gaussian functions, and so on, whereas only the trapezoidal membership functions [1] was allowed to use to prove the universal approximation theorem in Huwendiek and Brockmann [8]. It is mainly due to the fact that the THEN-parts of the fuzzy rules in the proposed HFS are, as shown in (6), composed of the terms of additions and multiplications of the outputs of the FLUs at the previous layer.

The following Stone–Weierstrass theorem is used to prove the universal approximation theorem of the proposed HFS.

Theorem 1 (Stone–Weierstrass theorem). *Let* $\mathbf{Z}$ *be a set of real continuous functions on a compact set* $\mathbf{X}$. *If* (1) $\mathbf{Z}$ *is an algebra, that is, the set* $\mathbf{Z}$ *is closed under addition, multiplication, and scalar multiplication*; (2) $\mathbf{Z}$ *vanishes at no point of* $\mathbf{X}$, *that is, for each* $\mathbf{x}\in\mathbf{X}$ *there exists* $f\in\mathbf{Z}$ *such that* $f(\mathbf{x})\neq 0$; (3) $\mathbf{Z}$ *separates points on* $\mathbf{X}$, *that is, for every* $\mathbf{x},\mathbf{x}'\in\mathbf{X},\mathbf{x}\neq\mathbf{x}'$, *there exist* $f\in\mathbf{Z}$ *such that* $f(\mathbf{x})\neq f(\mathbf{x}')$; *then the uniform closure of* $\mathbf{Z}$ *consists of all real continuous functions on* $\mathbf{X}$.

Proof. See [18]. □

Throughout this paper, let $\mathbf{F}$ be a set of all HFSs with the proposed structure, $\mathbf{U}\subset\mathbb{R}^n$ be a compact set of input variables denoted by $\mathbf{u}=(u_1,u_2,\ldots,u_n)^{\mathrm{T}}$, and $\mathbf{X}_{kd}\subset\mathbb{R}^{n(kd)}$ be a compact set of involved input variables denoted by $\mathbf{x}_{kd}=(x_1,x_2,\ldots,x_{n(kd)})^{\mathrm{T}}\subset\mathbf{u}$ for the dth FLU at the kth layer. m_j is the number of membership functions defined for x_j. We assume $n(kd)<n$ in this paper to construct a genuine HFS.

Lemma 1. $\mathbf{F}$ *is closed under addition.*

Proof. Let $f_1,f_2\in\mathbf{F}$ and construct an HFS as shown in Fig. 2. Let the number of hierarchical layers of the HFS be λ and, without loss of generality, the involved input variables for f_1 and f_2 be $\mathbf{x}_{(\lambda-1)1}$ and $\mathbf{x}_{(\lambda-1)2}$, respectively. The input variable in $\mathbf{x}_{\lambda 1}$ is then selected arbitrarily from those either in $\mathbf{x}_{(\lambda-1)1}$ or $\mathbf{x}_{(\lambda-1)2}$.

[1] It is the triangular membership functions that they are in the given compact domain.

If we set

$$\beta_j = \begin{cases} 0 & \text{if } (1 \leqslant j \leqslant n(\lambda 1)), \\ 1 & \text{if } (n(\lambda 1)+1 \leqslant j \leqslant n(\lambda 1)+2 = N(\lambda 1)), \end{cases}$$

$$\alpha_{uv} = 0 \quad \text{for all } (u,v) \in_{N(\lambda 1)} C_2,$$

$$\gamma = 0, \tag{7}$$

at the λth layer, then from (6) and (4), we have

$$y_{\lambda 1} = \varrho_{n(\lambda 1)+1} + \varrho_{n(\lambda 1)+2} = y_{(\lambda-1)1} + y_{(\lambda-1)2} = f_1 + f_2.$$

A choice of the coefficients of (3) that meet condition (7) is as follows:

$$\text{for all } 1 \leqslant i \leqslant \prod_{j=1}^{n(\lambda 1)} m_j,$$

$$p_j^i = \begin{cases} 0 & \text{if } (1 \leqslant j \leqslant n(\lambda 1)), \\ 1 & \text{if } (n(\lambda 1)+1 \leqslant j \leqslant n(\lambda 1)+2 = N(\lambda 1)), \end{cases}$$

$$w_{uv}^i = 0 \text{ for all } (u,v) \in_{N(\lambda 1)} C_2,$$

$$r^i = 0.$$

Since the HFS with the above configuration is in the form of the proposed HFS, **F** is closed under addition. □

Lemma 2. **F** *is closed under multiplication.*

Proof. In the same way as in the proof of Lemma 1, if we set

$$\beta_j = 0 \quad \text{for all } j \in \{1,2,\ldots,N(\lambda 1)\},$$

$$\alpha_{uv} = \begin{cases} 1 & \text{if } (u,v) = (n(\lambda 1)+1, n(\lambda 1)+2), \\ 0 & \text{otherwise}, \end{cases}$$

$$\gamma = 0, \tag{8}$$

then the output of the HFS follows from (6) and (4) that

$$y_{\lambda 1} = \varrho_{n(\lambda 1)+1}\varrho_{n(\lambda 1)+2} = y_{(\lambda-1)1}y_{(\lambda-1)2} = f_1 f_2.$$

A choice of the coefficients of (3) satisfying (8) is as follows:

$$\text{for all } 1 \leqslant i \leqslant \prod_{j=1}^{n(\lambda 1)} m_j,$$

$$p_j^i = 0 \quad \text{for all } j \in \{1,2,\ldots,N(\lambda 1)\},$$

$$w^i_{uv} = \begin{cases} 1 & \text{if}(u,v) = (n(\lambda 1)+1, n(\lambda 1)+2), \\ 0 & \text{otherwise,} \end{cases}$$

$$r^i = 0.$$

Since the HFS with the above configuration is in the form of the proposed HFS, **F** is closed under multiplication. □

Lemma 3. **F** *is closed under scalar multiplication.*

Proof. Let an arbitrary HFS with λ layer hierarchy be f_λ. From (5) and (6), we have

$$cf_\lambda = cy_{\lambda 1} = c\left\{\sum_{j=1}^{N(\lambda 1)} \beta_j \varrho_j + \sum_{(u,v)\in_{N(\lambda 1)}C_2} \alpha_{uv}\varrho_u\varrho_v + \gamma\right\}$$

$$= \sum_j \frac{\sum_i \mu^i(\mathbf{x}_{\lambda 1}) c p^i_j}{\sum_i \mu^i(\mathbf{x}_{\lambda 1})}\varrho_j + \sum_{(u,v)} \frac{\sum_i \mu^i(\mathbf{x}_{\lambda 1}) c w^i_{uv}}{\sum_i \mu^i(\mathbf{x}_{\lambda 1})}\varrho_u\varrho_v + \frac{\sum_i \mu^i(\mathbf{x}_{\lambda 1}) c r^i}{\sum_i \mu^i(\mathbf{x}_{\lambda 1})}$$

$$= \sum_j \frac{\sum_i \mu^i(\mathbf{x}_{\lambda 1}) \tilde{p}^i_j}{\sum_i \mu^i(\mathbf{x}_{\lambda 1})}\varrho_j + \sum_{(u,v)} \frac{\sum_i \mu^i(\mathbf{x}_{\lambda 1}) \tilde{w}^i_{uv}}{\sum_i \mu^i(\mathbf{x}_{\lambda 1})}\varrho_u\varrho_v + \frac{\sum_i \mu^i(\mathbf{x}_{\lambda 1}) \tilde{r}^i}{\sum_i \mu^i(\mathbf{x}_{\lambda 1})},$$

where

$$\tilde{p}^i_j = c p^i_j,$$

$$\tilde{r}^i = c r^i,$$

$$\tilde{w}^i_{uv} = c w^i_{uv}.$$

Since the HFS with above configuration is in the form of the proposed HFS, **F** is closed under scalar multiplication. □

Lemma 4. *For each* $\mathbf{u}\in\mathbf{U}$, *there exists* $f\in\mathbf{F}$ *such that* $f(\mathbf{u})\neq 0$, *i.e.*, **F** *vanishes at no point of* **U**.

Proof. We prove this by constructing the required f. Let $\mathbf{u} = (u_1, u_2, \ldots, u_n)^{\mathrm{T}}$.

Choose a HFS with only one FLU at the kth layer ($1\leqslant k\leqslant n$) as shown in Fig. 3 and set its ith fuzzy rule to

IF u_k is A^i_k THEN $\bar{y}^i_{k1}$ is $y_{(k-1)1} + C$,

where C is any nonzero constant.

Then, it follows that $y_{k1} = y_{(k-1)1} + C$.

Consequently, we have

$$f(\mathbf{u}) = y_{n1} = nC \neq 0 \quad \text{for all } \mathbf{u}\in\mathbf{U}$$

and **F** vanishes at no point of **U**. □

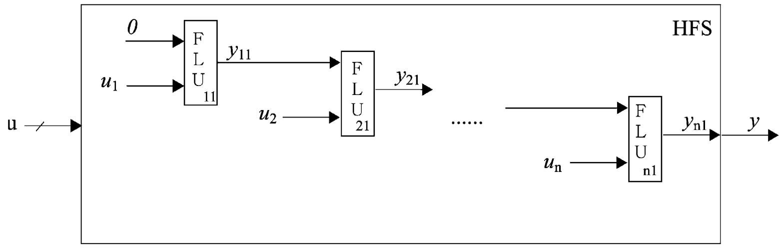


Fig. 3. HFS for Lemmas 4 and 5.

Lemma 5. *For every* $\mathbf{u}, \mathbf{u}' \in \mathbf{U}$ *and* $\mathbf{u} \neq \mathbf{u}'$, *there exists* $f \in \mathbf{F}$ *such that* $f(\mathbf{u}) \neq f(\mathbf{u}')$, *i.e.*, $\mathbf{F}$ *separates points on* $\mathbf{U}$.

Proof. We prove this by constructing the required f. Let $\mathbf{u} = (u_1, u_2, \ldots, u_n)^{\mathrm{T}}$ and $\mathbf{u}' = (u_1', u_2', \ldots, u_n')^{\mathrm{T}}$. Choose a HFS with only one FLU at the kth layer ($1 \leqslant k \leqslant n$) as shown in Fig. 3 and set its ith fuzzy rule to

$$\text{IF } u_k \text{ is } A_k^i \text{ THEN } \bar{y}_{k1}^i \text{ is } p_k u_k + y_{(k-1)1}.$$

Then, it follows that $y_{k1} = p_k u_k + y_{(k-1)1}$.

Consequently, we have

$$f(\mathbf{u}) = y_{n1} = \sum_{j=1}^{n} p_j u_j$$

and

$$f(\mathbf{u}) - f(\mathbf{u}') = \sum_{j=1}^{n} p_j \tilde{u}_j,$$

where $\tilde{u}_j = u_j - u_j'$.

If p_j is set to $\tilde{u}_j$, we have

$$f(\mathbf{u}) - f(\mathbf{u}') = \sum_{j=1}^{n} (\tilde{u}_j)^2 \neq 0 \quad \text{if } \mathbf{u} \neq \mathbf{u}'$$

and $\mathbf{F}$ separates points on $\mathbf{U}$. □

Theorem 2 (Universal approximation theorem). *For any continuous function* $g(\mathbf{u})$ *on a compact domain* $\mathbf{U}$ *and arbitrary* $\varepsilon > 0$, *there exists* $f \in \mathbf{F}$ *that satisfies*

$$\sup_{\mathbf{u} \in \mathbf{U}} |f(\mathbf{u}) - g(\mathbf{u})| < \varepsilon.$$

Proof. From (5) and (6), it is evident that $\mathbf{F}$ is a set of real continuous functions on $\mathbf{U}$ if $\sum_{i=1} \mu^i(\mathbf{x}_{kd}) > 0$ for all k and d, which are established by using complete fuzzy sets in the IF-parts of fuzzy rules. Using Lemmas 1, 2, and 3, $\mathbf{F}$ is proved to be an algebra. By using the Stone–Weierstrass theorem together with Lemmas 4 and 5, we establish that the proposed HFS possesses the universal approximation capability. □

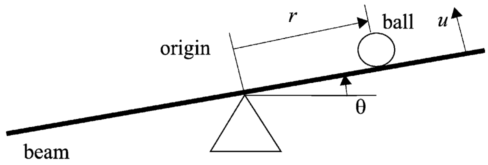


Fig. 4. The ball and beam system.

Remark 2. This theorem just guarantees the existence of an HFS that approximates a given continuous function. A reasonable HFS that can serve as a good approximator can be found by the partial knowledge of the plant [16,17,3,5,12] or by the structure searching routine such as genetic algorithms [11,20,19,9].

Remark 3. From (5), α, β, γ can be viewed as the outputs of the 0th order Takagi-Sugeno fuzzy logic systems (TS-FLSs). Since it is proved in [13,1,10] that TS-FLS has a universal approximation capability, α, β, γ can approximate any continuous functions on a compact set $\mathbf{X}_{kd}$.

4. Simulation

In the ball and beam system as shown in Fig. 4, let r be the distance of a ball from origin and θ be an angle that a beam makes with a horizontal line. Define $\mathbf{x} = (x_1, x_2, x_3, x_4)^{\mathrm{T}} = (r, \dot{r}, \theta, \dot{\theta})^{\mathrm{T}}$ as the state vector of the system and $y = r$ as the output of the system. The objective is to design an HFS with which the output converges to zero from arbitrary initial conditions in a certain region.

Representing the system with a state space model, we have

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} x_2 \\ B(x_1 x_4^2 - G \sin x_3) \\ x_4 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u$$

$$y = x_1$$

and based on this model, we choose a control law

$$u^*(\mathbf{x}) = \frac{4BGx_4 \cos x_3 + 6BG \sin x_3 - 4x_2 - x_1 - BGx_4^2 \sin x_3}{-BG \cos x_3} \tag{9}$$

from the input–output linearization algorithm of Hauser et al. [4], where B and G are 0.7143 and 9.81, respectively.

When this control law is applied to the system starting from four initial conditions $\mathbf{x}(0)=[2.4, -0.1, 0.6, 0.1]^{\mathrm{T}}$, $[1.6, 0.05, -0.6, -0.05]^{\mathrm{T}}$, $[-1.6, -0.05, 0.6, 0.05]^{\mathrm{T}}$, and $[-2.4, 0.1, -0.6, -0.1]^{\mathrm{T}}$, the corresponding trajectories are as shown in Fig. 5.

Granting that $u^*(\mathbf{x})$ is the ideal controller designed by using the full state model of the ball and beam system, we now try to design a fuzzy controller $\hat{u}(\mathbf{x})$ by using the proposed HFS, whose performance is comparable to that of $u^*(\mathbf{x})$. An HFS as shown in Fig. 6 may be not an optimal structure but is a natural choice, where two FLUs with inputs $(x_1, x_2)^{\mathrm{T}} = (r, \dot{r})^{\mathrm{T}}$ and $(x_3, x_4) = (\theta, \dot{\theta})^{\mathrm{T}}$ are placed at the 1st layer, and their outputs are inputs to the FLU at the 2nd layer to account for the dependency shown in (9). When three membership functions are defined for each input variable, the number of involved fuzzy rules becomes $3^2 + 3^2 + 3^2 = 27$ for the proposed HFS, but it is $3^4 = 81$ for a standard fuzzy logic system.

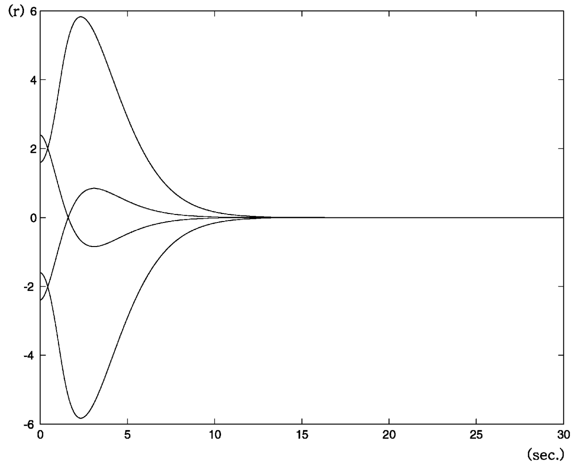


Fig. 5. Output $r(t)$ of the closed loop ball and beam system from four initial conditions when the input–output linearization algorithm of Hauser et al. is used.

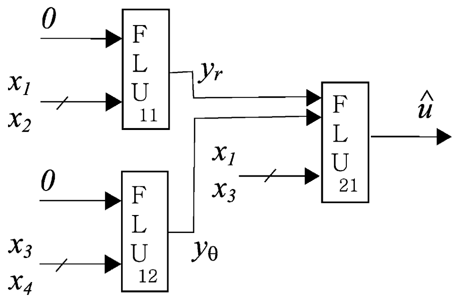


Fig. 6. A controller for ball and beam system.

In fact, when the terms in (6) are defined as; (a) for FLU_{11}, $\varrho_1 = x_1$, $\varrho_2 = x_2$, $\beta_1 = -1$, $\beta_2 = -4$, $\alpha_{12} = 0$, $\gamma = 0$; (b) for FLU_{12}, $\varrho_1 = x_3$, $\varrho_2 = x_4$, $\beta_1 = \beta_2 = 0$, $\alpha_{12} = 0$, $\gamma = 4BGx_4 \cos x_3 + 6BG \sin x_3 - BGx_4^2 \sin x_3$; and (c) for FLU_{21}, $\varrho_1 = x_1$, $\varrho_2 = x_3$, $\varrho_3 = y_{11}$, $\varrho_4 = y_{12}$, $\beta_1 = \beta_2 = 0$, $\beta_3 = \beta_4 = 1/-BG \cos x_3$, $\alpha_{uv} = 0$ for all (u, v), $\gamma = 0$, we have

$$y_{11} = -x_1 - 4x_2,$$

$$y_{12} = 4BGx_4 \cos x_3 + 6BG \sin x_3 - BGx_4^2 \sin x_3,$$

$$y_{21} = \hat{u}(\mathbf{x}) = u^*(\mathbf{x})$$

and an implementation of the fuzzy rules is given by

IF x_1 is A_1^i and x_2 is A_2^i THEN $\bar{y}_{11}^i$ is $p_1^i\varrho_1 + p_2^i\varrho_2$ for FLU_{11},

IF x_3 is A_3^i and x_4 is A_4^i THEN $\bar{y}_{12}^i$ is r^i for FLU_{12}, and

IF x_1 is A_1^i and x_3 is A_3^i THEN $\bar{y}_{21}^i$ is $p_3^i\varrho_3 + p_4^i\varrho_4$ for FLU_{21}.

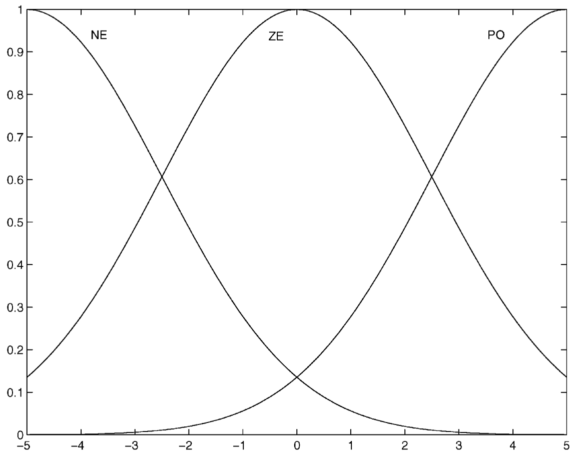


Fig. 7. Membership functions for r.

The parameters of the fuzzy rules used in the proposed HFS are updated by using the gradient descent algorithm with momentum, which is derived by minimizing the error criterion

$$J = \frac{1}{2}(u^* - u)^2.$$

Consequently, each parameter is updated as

$$h(k+1) = h(k) - \eta_1 \frac{\partial J}{\partial h} + \eta_2 \Delta h(k),$$

$$\Delta h(k) = h(k) - h(k-1),$$

where η_1 and η_2 are adaptation gains. The parameters to be updated are the coefficients of the THEN-parts of fuzzy rules. Random numbers from -1 to 1 are assigned initially for these parameters and η_1 and η_2 are set to 0.005 and 0.0005, respectively.

The ranges of x_1, x_2, x_3, and x_4 are set to $[-5,\ 5]$, $[-2,\ 2]$, $[-\pi/4,\ \pi/4]$, and $[-0.8,\ 0.8]$, respectively, and target samples consist of 300 input–output pairs which are generated from (9) with randomly selected inputs in the given ranges.

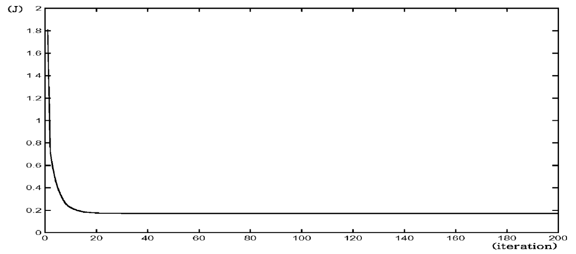
The FLUs used in the simulation are the ones with singleton fuzzifier, product inference, and center average defuzzifier. Input fuzzy set is characterized by the gaussian membership function of the form

$$f_m(x) = \mathrm{e}^{-(x-x_m)^2/2\sigma^2},$$

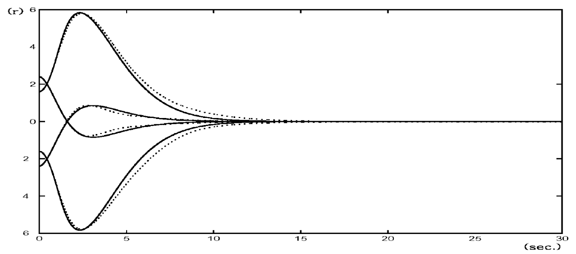
where x_m is equally spaced and σ is set to the half of the length between adjacent x_m's. Fig. 7 shows an example when three membership functions are used for r with NE, ZE, and PO representing *negative*, *zero*, and *positive*, respectively.

After updating the coefficients of the THEN-parts, we have simulated the developed controller with 100 ms sampling time starting from four different initial conditions. Fig. 8 shows that its performance is comparable to that of $u^*(\mathbf{x})$ in (9) and the proposed HFS approximates with good accuracy the nonlinear control input $u^*(\mathbf{x})$ with just 27 fuzzy rules.

Tables 1 and 2 are the resulting fuzzy rules.



(a) Learning plot ($J = \frac{1}{300}\sqrt{\sum_{k=1}^{300}(u_k^* - \hat{u}_k)^2}$)



(b) $r(t)$ of the closed loop ball and beam system (solid : results with the input-output linearization algorithm, dotted : results with the proposed controller)

Fig. 8. Learning plot and output $r(t)$ of the closed loop ball and beam system. (a) Learning plot ($J = \frac{1}{300}\sqrt{\sum_{k=1}^{300}(u_k^* - \hat{u}_k)^2}$) (b) $r(t)$ of the closed loop ball and beam system (solid: results with the input–output linearization algorithm, dotted: results with the proposed controller).

Table 1
Fuzzy rules at the 1st layer after learning ($(x_1, x_2, x_3, x_4)^T = (r, \dot{r}, \theta, \dot{\theta})^T$)

	IF		THEN			IF		THEN
	x_1	x_2	p_1	p_2		x_3	x_4	r
FLU for y_r (y_{11})	NE	NE	−0.932089	−0.656577	FLU for y_θ (y_{12})	NE	NE	−3.669430
	NE	ZE	−0.427016	−0.330597		NE	ZE	−2.605983
	NE	PO	0.146333	−0.459912		NE	PO	−0.776843
	ZE	NE	−0.091973	−0.807471		ZE	NE	−1.545487
	ZE	ZE	−0.158190	−0.811557		ZE	ZE	−0.005481
	ZE	PO	−0.083643	−0.854756		ZE	PO	1.530980
	PO	NE	−0.089672	−0.620114		PO	NE	0.790945
	PO	ZE	−0.293712	−0.649295		PO	ZE	2.626142
	PO	PO	−0.553821	−0.578318		PO	PO	3.768389

Table 2
Fuzzy rules at the 2nd layer after learning

	IF		THEN	
	x_1	x_3	p_3	p_4
	NE	NE	−0.392772	−2.568475
	NE	ZE	−0.260173	−1.558061
FLU for	NE	PO	−0.338405	−2.516689
	ZE	NE	−1.174745	−2.599592
$\hat{u}\ (y_{21})$	ZE	ZE	−0.721210	−1.474677
	ZE	PO	−1.218626	−2.556902
	PO	NE	−0.607635	−2.626210
	PO	ZE	−0.427968	−1.494636
	PO	PO	−0.598062	−2.559922

5. Conclusion

The conventional HFS does reduce the number of fuzzy rules dramatically, but it is difficult to design because intermediate output variables with less or no physical meaning are used as input variables for the FLUs at the next layer. In contrast, the proposed HFS uses only the original input variables with clear physical meaning in all of the IF-parts of fuzzy rules, thereby rendering the involved fuzzy rules easy to interpret and design.

Any real continuous functions on a compact set are proved to be approximated by the proposed HFS to any degree of accuracy. From the simulation of ball and beam control system, it is demonstrated that the proposed scheme approximates with good accuracy a nonlinear controller with fewer fuzzy rules than the centralized fuzzy system and its control performance is comparable to that of a model nonlinear controller.

References

[1] J.J. Buckley, Yoichi Hayashi, Fuzzy input–output controllers are universal approximators, Fuzzy Sets and Systems 58 (1993) 273–278.
[2] Chen Wei, Li-Xin Wang, A note on universal approximation by hierarchical fuzzy systems, Inform. Sci. 123 (2000) 241–248.
[3] Fu-lai Chung, Ji-cheng Duan, Daniel So Yeung, Deriving multistage FNN models from Takagi and Sugeno's fuzzy systems, IEEE Internat. Conf. on Fuzzy Systems, 1998, pp. 1259–1264.
[4] J. Hauser, S. Sastry, P. Kokotovic, Nonlinear control via approximate input–output linearization: the ball and beam example, IEEE Trans. Automat. Control AC-37 (1992) 392–398.
[5] R. Holve, Investigation of automatic rule generation for hierarchical fuzzy systems, IEEE Trans. Systems, Man Cybernet. — part C: Applications and Reviews 29 (1999) 98–109.
[6] O. Huwendiek, W. Brockmann, NetFAN — a structured adaptive fuzzy approach, IEEE Internat. Conf. on Neural Networks, Piscataway 1996, pp. 1079–1084.
[7] O. Huwendiek, W. Brockmann, On the applicablility of the NetFAN-approach to function approximation, IEEE Internat. Conf. on Fuzzy Systems, Piscataway 1997, pp. 477–482.
[8] O. Huwendiek, W. Brockmann, Function approximation with decomposed fuzzy systems, Fuzzy Sets and Systems 101 (1999) 273–286.
[9] Kanta Tachibana, Takeshi Furuhashi, Uneven allocation of membership functions for hierarchical fuzzy modeling using genetic algorithm, IEEE Internat. Conf. on Evolutionary Computation, Alaska, 1998, pp. 746–751.
[10] Ke Zeng, Nai-Yao Zhang, Wen-Li Xu, A comparative study on sufficient conditions for Takagi-Sugeno fuzzy systems as universal approximators, IEEE Trans. Fuzzy Systems 8 (2000) 773–780.
[11] Koji Shimojima, Toshio Fukuda, Yasuhisa Hasegawa, Self-tuning fuzzy modeling with adaptive membership function, rules, and hierarchical structure based on genetic algorithm, Fuzzy Sets and Systems 71 (1995) 295–309.

[12] D.A. Linkens, H.O. Nyongesa, A hierarchical multivariable fuzzy controller for learning with genetic algorithms, Internat. J. Control 63 (1996) 865–883.
[13] Li-Xin Wang, A Course in Fuzzy Systems and Control, Prentice-Hall PTR, Englewood Cliffs, NJ, 1997.
[14] Li-Xin Wang, Universal approximation by hierarchical fuzzy systems, Fuzzy Sets and Systems 93 (1998) 223–230.
[15] Li-Xin Wang, Analysis and design of hierarchical fuzzy systems, IEEE Trans. Fuzzy Systems 7 (1999) 617–624.
[16] G.V.S. Raju, J. Zhou, R.A. Kisner, Hierarchical fuzzy control, Internat. J. Control 54 (1991) 1201–1216.
[17] G.V.S. Raju, Jun Zhou, Adaptive hierarchical fuzzy controller, IEEE Trans. Systems, Man Cybernet. 23 (1993) 973–980.
[18] W. Rudin, Principles of Mathematical Analysis, McGraw-Hill, New York, 1976.
[19] Takeshi Furuhashi, Seiichi Matsushita, Hiroaki Tsutsui, Yoshiki Uchikawa, Knowledge extraction from hierarchical fuzzy model obtained by fuzzy neural networks and genetic algorithm, Internat. Conf. on Neural Networks, Vol. 4, Texas, 1997, pp. 2374–2379.
[20] Toshio Fukuda, Yasuhisa Hasegawa, Koji Shimojima, Structure organization of hierarchical fuzzy model using by genetic algorithm, IEEE Internat. Conf. on Fuzzy Systems, Vol. 1, 1995, pp. 295–300.