Some Properties of the Fuzzy Rule Table for Polynomials of two Variables

Jeong A Ryou*, Sei Young Chung*, Byung Soo Moon**

* Dept. of Mathematics, ChungNam National University
220 Gung-Dong Yuseong-Gu, Taejon, KOREA 305-600
** Korea Atomic Energy Research Institute
Tel : (011)456-8520 Fax : (042)486-9885 E-mail : Master@JeongA.com

Abstract

In this paper, we consider a fuzzy system representation for polynomials of two variables. The representation we use is an exact transformation of the corresponding cubic spline interpolation function. We examine some of the properties of their fuzzy rule tables and prove that the rule table is symmetric or antisymmetric depending on whether the polynomial is symmetric or antisymmetric. A few examples are included to verify our proof. These results not only provide some insights on properties of the cubic spline interpolation coefficients but also provide some help in setting up fuzzy rule tables for functions of two variables.

1. Introduction

It is well known that fuzzy systems can be used to approximate continuous functions on a compact set within arbitrary accuracy[1]. There are several papers describing methods on how to construct fuzzy systems within a prescribed accuracy[3], one of which is based on the cubic spline interpolation. Let $f(x)$ be a continuously differentiable function on $[-1\ \ 1]$. We divide the interval $[-1,1]$ into $2n$ sub intervals and let $x_j=-1+jh,\ j=-3,-2,-1,\cdots,2n+3$ with $h=\frac{1}{n}$. The cubic B-spline function[2] defined on $[x_{i-2},x_{i+2}]$ is

$$B_{i(x)}=\frac{1}{6h^3}\times\begin{cases}(x-x_{i-2})^3 & \text{if } x\in[x_{i-2},x_{i-1}]\\ h^3+3h^2(x-x_{i-1})+3h(x-x_{i-1})^2-3(x-x_{i-1})^3 & \text{if } x\in[x_{i-1},x_i]\\ h^3+3h^2(x_{i+1}-x)+3h(x_{i+1}-x)^2-3(x_{i+1}-x)^3 & \text{if } x\in[x_i,x_{i+1}]\\ (x_{i+2}-x)^3 & \text{if } x\in[x_{i+1},x_{i+2}]\\ 0 & \textit{otherwise}\end{cases}$$

for $i=-1,0,1,\cdots,2n+1$.

The fuzzy system representing the cubic spline interpolation of a given function is formed by the following four steps[4].

(1) Fuzzy set for input fuzzification: We take $B_i(x)$ for $i=-n-1,\cdots,0,\cdots,n+1$ as fuzzy set for fuzzifying the input value x.

(2) Generation of fuzzy rules: We sort $(2n+3)$ coefficients c_i in an increasing order and delete the duplicate ones. Assign the ordinal number 1 to the smallest c_i and 2 to the second smallest and so forth. The largest c_i will have ordinal number N which is less than or equal to $2n+3$. We then form the rule vector R so that $i-th$ entry r_i is the ordinal (fuzzy set) number corresponding to coefficient c_i.

(3) Output fuzzy set: Let $\{t_k \mid k=1,2,\cdots,N\}$ be the sorted array of c_i's. For each k, define an output fuzzy set T_k to be spike function whose support is $[t_{k-1},t_{k+1}]$.

(4) Defuzzification: For the defuzzification of

the output, we use the center area defuzzification method.

In the following, we will prove that the fuzzy rule table are either symmetric or antisymmetric depending on whether the function is symmetric or antisymmetric.

2. Some properties of the fuzzy rules

Lemma 1. Let $f(x)$ be a continuously differentiable function on the interval [-1, 1]. We divide the interval into $2n$ subintervals and let $x_i = \frac{i}{n} = ih$ with $h = \frac{1}{n}$. Let $S(x)$ be the cubic spline interpolation of f(x) of the form $\sum_{i=-n-1}^{n+1} c_i B_i(x)$, where $B_i(x)$ is the cubic B-spline as defined above. Let C be the vector with the $2n+3$ coefficients $c_i's$ as entries and let R be the rule vector such that the ith entry r_i is fuzzy set number corresponding to the coefficient c_i.

(1) If $f(x)$ is an even function, then we have
$c_i = c_{-i},\ r_i = r_{-i} \quad i = 1, 2, \cdots, n+1.$
(2) If $f(x)$ is an odd function, then we have
$c_i = -c_{-i},\ r_i + r_{-i} = 2r_0 \quad i = 1, 2, \cdots, n+1.$
where $c_0 = 0$, r_0 is the center of ordinal numbers.

Proof. To compute the interpolation coefficients of $f(x)$, we must solve a set of linear equations for the interpolation constraints $s'(x_{-n}) = f'(x_{-n})$, $s(x_i) = f(x_i)$, for $-n \le i \le n$, $s'(x_n) = f'(x_n)$.

The matrix equation involved is $Ax = b$, where,

$$x = [c_{-n-1},\ c_{-n},\ \cdots,\ c_0,\ \cdots,\ c_n,\ c_{n+1}]^T,$$

$$b = [f'(x_{-n}), f(x_{-n}), f(x_{-n+1}), \cdots, f(x_0), \cdots, f(x_{n-1}), f(x_n), f'(x_n)]^T$$

and the coefficient matrix.
A is given by

$$A = \frac{1}{6}\begin{bmatrix} -\frac{3}{h} & 0 & \frac{3}{h} & 0 & \cdots & 0 \\ 1 & 4 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 4 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & 1 & 4 & 1 \\ \cdots & \cdots & 0 & \frac{-3}{h} & 0 & \frac{3}{h} \end{bmatrix}$$

The matrix A is non-singular since A is diagonally dominant and hence the linear matrix equation $Ax = b$ has the unique solution. Then the cubic spline interpolation function for $f(x)$ can be written as

$$S(x) = \sum_{i=-n-1}^{n+1} c_i B_i(x).$$

We let $\tilde{S}(x) = S(-x) = \sum_{i=-n-1}^{n+1} c_i B_i(-x)$ and then $\tilde{S}'(x) = \sum_{i=-n-1}^{n+1} c_i (B_i(-x))'$

$$= \sum_{i=-n-1}^{n+1} -c_i B_i'(-x). .$$

$\tilde{S}'(x)$ at the both ending node x_{-n} and x_n become

$$\tilde{S}'(x_{-n}) = \sum_{i=-n-1}^{n+1} -c_i B_i'(-x_{-n})$$

$$= \sum_{i=-n-1}^{n+1} -c_i B_i'(x_n)$$

$$= -\frac{1}{2}h \cdot -c_{n-1} + \frac{1}{2h} \cdot -c_{n+1}$$

$$= -f'(-x_n) = (f(-x_{-n}))'$$

$$\tilde{S}'(x_n) = \sum_{i=-n-1}^{n+1} -c_i B_i'(-x_n)$$

$$= \frac{1}{2h} \cdot c_{n-1} - \frac{1}{2h} \cdot c_{-n+1} = (f(-x_n))'$$

Since $[-1 \quad 1]$ is used for the interval and $2n$ subintervals are used, we have $x_{-i} = -x_i$ for all $i = 1, 2, \cdots, n$. Now, consider the value of $\tilde{S}(x)$ at the node x_j, $-n \le j \le n$.

$$\tilde{S}(x_j) = \sum_{i=-n-1}^{n+1} c_i B_i(-x_j)$$

$$= \sum_{i=-n-1}^{n+1} c_i B_i(x_{-j})$$

$$= c_{-j-1}B_{-j-1}(x_{-j}) + c_{-j}B_{-j}(x_{-j}) + c_{-j+1}B_{-j+1}(x_{-j})$$
$$= \frac{1}{6}c_{-j-1} + \frac{4}{6}c_{-j} + \frac{1}{6}c_{-j+1}$$
$$= f(x_{-j}) = f(-x_j),\ -n \le j \le n.$$

Therefore, we have

$$\tilde{S}'(x_{-n}) = (f(-x_{-n}))',$$
$$\tilde{S}(x_j) = f(-x_j),\ -n \le j \le n,$$
$$\tilde{S}'(x_n) = (f(-x_n))',$$

from which we conclude that $\tilde{S}(x)$ is the spline interpolation function of $f(-x)$. Now note that

$$\tilde{S}(x) = S(-x) = \sum_{i=-n-1}^{n+1} c_i B_i(-x)$$
$$= \sum_{i=-n-1}^{n+1} c_i B_{-i}(x)$$
$$= \sum_{i=-n-1}^{n+1} c_{-i} B_i(x).$$

If $f(x) = f(-x)$ for all $x \in [-1\ \ 1]$, then the spline interpolation functions of $f(x)$ and $f(-x)$ are the same, and hence we must have $c_i = c_{-i}$ and therefore $r_i = r_{-i}$, for all $i = 1, 2, \cdots, n$. A similar proof for the odd functions is omitted.

Example 1.
The spline interpolation coefficients of $f(x) = x^2$ and the ordinal (fuzzy set) numbers are as shown in table 1.

Table 1.

Ci VALUE of f(x) = x²		**Fuzzy Rule f(x) = x²**
C-3 =	8.666667	4.0
C-2 =	3.666667	3.0
C-1 =	6.666667E-01	2.0
C0 =	-3.333333E-01	1.0
C1 =	6.666667E-01	2.0
C2 =	3.666667	3.0
C3 =	8.666667	4.0

Theorem 1. Let $f(x,y)$ be a polynomial of two variables x, y and let $S(x,y)$ be the cubic spline interpolation of the form $\sum_{i,j=-n-1}^{n+1} a_{ij} B_i(x) B_j(y)$ on the interval $[-1,1]\times[-1,1]$ where $B_i(x)$ and $B_j(y)$ are the cubic B-splines defined above. If A is the $(2n+3)\times(2n+3)$ matrix of the coefficients a_{ij}, and if R is a $(2n+3)\times(2n+3)$ rule table matrix such that r_{ij} is the ordinal (fuzzy set) number corresponding to the coefficient a_{ij} in the set of number $\{a_{kl} \mid k, l = -n-1, -n, \cdots, n+1\}$, then we have the following.

i) If $f(-x,y) = f(x,y)$ for all $x, y \in [-1,1]$, then $a_{ij} = a_{(-i),j}$, $r_{ij} = r_{(-i),j}$

ii) If $f(x,-y) = f(x,y)$ for all $x, y \in [-1,1]$, then $a_{ij} = a_{i,(-j)}$, $r_{ij} = r_{i,(-j)}$

iii) If $f(-x,-y) = f(x,y)$ for all $x, y \in [-1,1]$, then $a_{ij} = a_{(-i),(-j)}$, $r_{ij} = r_{(-i),(-j)}$

iv) If $f(-x,y) = -f(x,y)$ for all $x, y \in [-1,1]$, then $a_{ij} = -a_{ij}$, $a_{o,j} = 0$, $r_{ij} + r_{(-i),j} = 2r_{0,j}$

v) If $f(x,-y) = -f(x,y)$ for all $x, y \in [-1,1]$, then $a_{i,(-j)} = -a_{ij}$, $a_{i,0} = 0$, $r_{ij} + r_{i,(-j)} = 2r_{i,0}$

vi) If $f(-x,-y) = -f(x,y)$ for all $x, y \in [-1,1]$, then $a_{(-i),(-j)} = -a_{ij}$, $a_{ij} = 0$. $r_{ij} + r_{(-i),(-j)} = 2r_{0,0}$

where, $r_{0,j}, r_{i,0}$ and $r_{0,0}$ are the center of rank $r_{ij}'s$.

Proof. Let $f_k(x) = x^k$, $f_l(y) = y^l$, $f_{kl}(x,y) = f_k(x)f_l(y) = x^k y^l$ and

$$f(x,y) = \sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl} x^k y^l.$$

If $S_k(x) = \sum_{i=-n-1}^{n+1} c_i^{(k)} B_i(x)$ and $S_l(y) = \sum_{i=-n-1}^{n+1} d_j^{(l)} B_j(y)$ are the spline interpolation function of $f_k(x)$ and $f_l(y)$ respectively, then the spline interpolation functions for $f(x, y) = \sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}x^k y^l$ $= \sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}f_k(x) f_l(y)$ can be written as

$$S(x, y) = \sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}S_k(x) S_l(y)$$
$$= \sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}\left(\sum_{i=-n-1}^{n+1}\sum_{j=-n-1}^{n+1} c_i^{(k)}d_j^{(l)}B_i(x)B_j(y)\right)$$
$$= \sum_{i=-n-1}^{n+1}\sum_{j=-n-1}^{n+1}\left(\sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}c_i^{(k)}d_j^{(l)}B_i(x)B_j(y)\right)$$
$$= \sum_{i=-n-1}^{n+1}\sum_{j=-n-1}^{n+1} a_{ij}B_i(x)B_j(y)$$

where $a_{ij} = \sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}c_i^{(k)}d_j^{(l)}$.

The spline interpolation function for $f(-x, y) = \sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}f_k(-x) f_l(y)$ becomes

$$S(-x, y) = \sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}S_k(-x) S_l(y)$$

by lemma 1.

$$= \sum_{i=-n-1}^{n+1}\sum_{j=-n-1}^{n+1}\left(\sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}c_i^{(k)}d_j^{(l)}B_i(-x)B_j(y)\right)$$
$$= \sum_{i=-n-1}^{n+1}\sum_{j=-n-1}^{n+1}\left(\sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}c_i^{(k)}d_j^{(l)}B_{-i}(x)B_j(y)\right)$$
$$= \sum_{i=-n-1}^{n+1}\sum_{j=-n-1}^{n+1}\left(\sum_{k=0}^{n}\sum_{l=0}^{m} e_{kl}c_{-i}^{(k)}d_j^{(l)}B_i(x)B_j(y)\right)$$
$$= \sum_{i=-n-1}^{n+1}\sum_{j=-n-1}^{n+1} a_{-ij}B_i(x)B_j(y)$$

In case i)

Since the spline interpolation function $S(x, y)$ for $f(x,y)$ is equal to the spline interpolation function $S(-x,y)$ for $f(-x,y)$, hence $a_{ij} = a_{-ij}$, therefore $r_{ij} = r_{-ij}$.

ii) ~ Vi) are similar to I).

Example 2.

The coefficients of $f(x, y) = x^2y^2$ and the fuzzy rule are as shown in table 2.

Table 2.

C_{ij} VALUE of f(x,y) = x^2y^2							Fuzzy Rule to Compute f(x,y) = x^2y^2						
17.	12.	9.	8.	9.	12.	17.	10.	9.	8.	7.	8.	9.	10.
12.	7.	4.	3.	4.	7.	12.	9.	6.	5.	4.	5.	6.	9.
9.	4.	1.	0.	1.	4.	9.	8.	5.	3.	2.	3.	5.	8.
8.	3.	0.	.6	0.	3.	8.	7.	4.	2.	1.	2.	4.	7.
9.	4.	1.	0.	1.	4.	9.	8.	5.	3.	2.	3.	5.	8.
12.	7.	4.	3.	4.	7.	12.	9.	6.	5.	4.	5.	6.	9.
17.	12.	9.	8.	9.	12.	17.	10.	9.	8.	7.	8.	9.	10.

Theorem 2.

If $f(x,y) = f(y,x)$ then $a_{ij} = a_{ji}$, $r_{ij} = r_{ji}$

if $f(x,y) = f(-y,-x)$ then $a_{ij} = a_{-j-i}$, $r_{ij} = r_{-j-i}$

if $f(x,y) = f(-y,x)$ then $a_{ij} = a_{-ji}$, $r_{ij} = r_{-j-i}$

if $f(x,y) = f(y,-x)$ then $a_{ij} = a_{j-i}$, $r_{ij} = r_{j-i}$

if $-f(x,y) = f(y,x)$

then $a_{ij} = -a_{ji}$, $a_{ii} = 0$, $r_{ij} + r_{ji}\ 2r_{ii}$

3. Conclusion

We showed that for functions with symmetric or antisymmetric properties, only a part of the fuzzy rule need be computed. So we can obtain the whole of rest coefficients and fuzzy rules from the portion of the coefficients. It can be shown that a function is symmetric or antisymmetric relative to an arbitrary point, then the fuzzy rules are symmetric or antisymmetric.

Reference.

[1] J.L.Castro, M.Delgado, Fuzzy system with defuzzification are universal approximators, IEEE Trans. systems man cybernet. 26(1) (1996) 149-152.

[2] P.M.Prenter, Splines and Variational Methods, Wiley, NewYork, 1975.

[3] P.Wang et al., Constructive theory for fuzzy sustems, Fuzzy set and systems 88(1997) 195-203.

[4] B.S. Moon, A practical algorithm for representing polynomials of two variable by fuzzy systems with accuracy O(h^4), Fuzzy sets and systems, 1999.