The Third AFSS(1998), 562-567

Relations among the multidimensional linear interpolation, fuzzy reasoning, and neural networks

*Kyong-Sik Om, Hee-Chan Kim†, and Byoung-Goo Min†

*Dept. of Biomedical Engineering, College of Engineering, Seoul National University Hospital,
28 Yongon-dong Chongno-gu, Seoul 100-744, Korea
Tel:+82-02-760-3128, 3125, Fax: +82-02-745-7870, E-mail: ksom@biomed.snu.ac.kr

†Dept. of Biomedical Engineering, College of Medicine, Seoul National University Hospital
28 Yongon-dong Chongno-gu, Seoul 100-744, Korea
Tel:+82-02-760-3125, Fax: +82-02-745-7870,
E-mail: hckim@biomed.snu.ac.kr and bgmin@biomed.snu.ac.kr

Abstract

This paper examined the relations among the multidimensional linear interpolation (MDI) and fuzzy reasoning, and neural networks, and showed that an MDI is a special form of Tsukamoto's fuzzy reasoning and regularization networks in the perspective of fuzzy reasoning and neural networks, respectively. For this purposes, we proposed a special Tsukamoto's membership (STM) system and triangular basis function (TBF) network. Also we verified the condition when our proposed TBF becomes a well-known radial basis function (RBF).

Keywords: Multidimensional linear interpolation (MDI), fuzzy reasoning, neural networks, regularization networks.

1. Introduction

The training process of a neural network may be viewed as one of curve fitting [1]. Interpolation technique is used in the application of signal processing [2], fuzzy learning [3] and so on. Multidimensional linear interpolation (MDI) is a useful method for nonlinear function problem. This paper examined the relations among the MDI, fuzzy reasoning, and neural networks, and showed that an MDI is a special form of Tsukamoto's fuzzy reasoning [4][5] and regularization networks [6] in the perspective of fuzzy reasoning and neural networks, respectively. For this purpose, we proposed a special Tsukamoto's membership (STM) system and triangular basis function (TBF) networks. Also we verified the condition when our proposed TBF becomes a well-known radial basis function (RBF).

This paper is organized as follows. We stated an MDI in section 2. In section 3, we derived the MDI from the STM system. In section 4, we derived the MDI from the proposed TBF networks. In section 5, we summarized and discussed about our studies. Finally, in section 6, conclusions were stated.

2. Multidimensional Linear Interpolation

Before we proceed, it is necessary to comprehend that what we mean the MDI is the problem of interpolating on a mesh that is Cartesian, i.e., has not tabulated function values at 'random' points in n-dimensional space rather than at the vertices of a rectangular array. This rectangular data array will be called a look-up table (LUT) from now. For simplicity, we consider only the case of three dimensions, the cases of two and four or more dimensions being analogous in every way. If the input variable arrays are $x_{1a}[\ ]$, $x_{2a}[\ ]$, and $x_{3a}[\ ]$, the output y (x_1, x_2, x_3) has following relation [7].

$$y_a[m][n][r] = y(x_{1a}[m], x_{2a}[n], x_{3a}[r]). \tag{1}$$

The goal is to estimate, by interpolation, the function y at some untabulated point (x_1, x_2, x_3). If x_1, x_2, x_3 satisfy

$$\begin{cases} x_{1a}[\ m\ ] \le x_1 \le x_{1a}[\ m+1\ ] \\ x_{2a}[\ n\ ] \le x_2 \le x_{2a}[\ n+1\ ] \\ x_{3a}[\ r\ ] \le x_3 \le x_{3a}[\ r+1\ ], \end{cases} \tag{2}$$

the grid points are

$$\begin{aligned} y_1 &= y_a[\ m\ ][\ n\ ][\ r\ ], \\ y_2 &= y_a[\ m\ ][\ n\ ][\ r+1\ ], \\ y_3 &= y_a[\ m\ ][\ n+1\ ][\ r\ ], \\ y_4 &= y_a[\ m\ ][\ n+1\ ][\ r+1\ ], \\ y_5 &= y_a[\ m+1\ ][\ n\ ][\ r\ ], \\ y_6 &= y_a[\ m+1\ ][\ n\ ][\ r+1\ ], \\ y_7 &= y_a[\ m+1\ ][\ n+1\ ][\ r\ ], \\ y_8 &= y_a[\ m+1\ ][\ n+1\ ][\ r+1\ ]. \end{aligned} \tag{3}$$

The final 3-dimensional linear interpolation is

$$\begin{aligned} y(x_1\ ,\ x_2\ ,\ x_3\ ) = & \\ & (1-u)(1-v)(1-w)y_1 \\ &+ (1-u)(1-v)(w)y_2 \\ &+ (1-u)(v)(1-w)y_3 \\ &+ (1-u)(v)(w)y_4 \\ &+ (u)(1-v)(1-w)y_5 \\ &+ (u)(1-v)(w)y_6 \\ &+ (u)(v)(1-w)y_7 \\ &+ (u)(v)(w)y_8, \end{aligned} \tag{4}$$

where

$$\begin{aligned} u &= \frac{x_1 - x_{1a}[m]}{x_{1a}[m+1] - x_{1a}[m]}, \\ v &= \frac{x_2 - x_{2a}[n]}{x_{2a}[n+1] - x_{2a}[n]}, \\ w &= \frac{x_3 - x_{3a}[r]}{x_{3a}[r+1] - x_{3a}[r]}. \end{aligned} \tag{5}$$

(u, v, and w each lie between 0 and 1.)

We can see the estimated y uses 2^n table terms if n-dimensions, and it satisfies 8 terms in the case of three dimensions as above.

3. Tsukamoto's Fuzzy Reasoning

Tsukamoto used monotonic membership functions for linguistic terms [4]. As an example, consider the case of two input variables and one output variable.

$$\begin{aligned} &R1 : \mathit{If}\ x_1 = A_{11} \text{ and } x_2 = A_{21}, \text{ then } y_1 = B_1, \\ &R2 : \mathit{If}\ x_1 = A_{12} \text{ and } x_2 = A_{22}, \text{ then } y_2 = B_2, \\ &R3 : \mathit{If}\ x_1 = A_{13} \text{ and } x_2 = A_{23}, \text{ then } y_3 = B_3, \\ &R4 : \mathit{If}\ x_1 = A_{14} \text{ and } x_2 = A_{24}, \text{ then } y_4 = B_4. \end{aligned} \tag{6}$$

where

y_i: Inferred variable of the consequence.
x_1, x_2: Variables of the premise.
A_{1i}, A_{2i}: Normalized fuzzy sets over the input domain U and V.
B_i: Normalized fuzzy sets over the output domain W.

If we define the fuzzified value A_1' and A_2' for input $x_1 = x_1^0$, $x_2 = x_2^0$, as fuzzy singletons as follows,

$$\begin{aligned} A_1' &= \begin{cases} 1, & \text{if } x_1 = x_1^0, \\ 0, & \text{otherwise}, \end{cases} \\ A_2' &= \begin{cases} 1, & \text{if } x_2 = x_2^0, \\ 0, & \text{otherwise}, \end{cases} \end{aligned} \tag{7}$$

the compatibility w_i for ith rule is,

$$w_i = A_{1i}(x_1^0) \wedge A_{2i}(x_2^0), \tag{8}$$

or

$$w_i = A_{1i}(x_1^0) A_{2i}(x_2^0). \tag{9}$$

Here, Eq. (8) means logical product ($\wedge$), and Eq. (9) algebraic product (•). The result y_i^* inferred from R_i is defined as follow :

$$w_i = B(y_i^*) \rightarrow y_i^* = B^{-1}(w_i). \tag{10}$$

The final inferred value y^* from all rules usually calculated by *weighted combination method* as follows :

$$y^* = \frac{\sum_{i=1}^{4} w_i B_i^i(w_i)}{\sum_{i=1}^{4} w_i} \tag{11}$$

B_i must be monotonic, whereas A_{1i}, and A_{2i} have no restriction of shape.

3.1 Expression of multidimensional linear interpolation from fuzzy reasoning

Triangular membership functions are used to subdivide the input universe. A fuzzy set A_i defined by triangular membership functions has the form

$$\mu(x) = \begin{cases} \dfrac{x - a_{i-1}}{a_i - a_{i-1}}, & \text{if } a_{i-1} \le x \le a_i \\ \dfrac{-x + a_{i+1}}{a_{i+1} - a_i}, & \text{if } a_i \le x \le a_{i+1} \\ 0, & \text{otherwise}. \end{cases} \tag{12}$$

The point a_i will be referred to as the midpoint of A_i. The leftmost and rightmost fuzzy regions are truncated with the midpoint as the leftmost and rightmost position, respectively. And the straight line membership functions are used for the output universe W. Fig. 1 and Fig. 2 help to understand the above relations. From now, previous membership system will be called a STM (Special Tsukamoto's Membership) system.

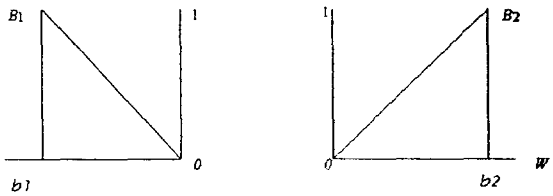


Fig. 1. Triangular decomposition of inputdomain.

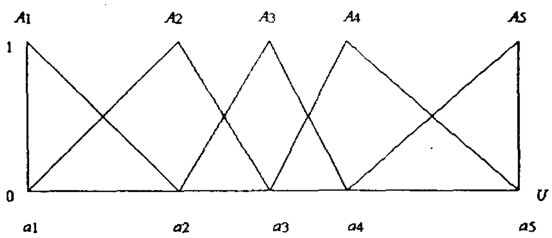


Fig. 2. Representing output domain usingstraight line monotonic membership function.

As an example, we consider three input variables. If we use fuzzy singletons as inputs x_1^0, x_2^0 and x_3^0, the compatibility w_i is $A_{1i}(x_1^0) \nabla A_{2i}(x_2^0) \nabla A_{3i}(x_3^0)$ where ∇ denotes triangular norms (T-norm). And $w_i = 0$ if $A_{1i}(x_1^0) = 0$ or $A_{2i}(x_2^0) = 0$ or $A_{3i}(x_3^0) = 0$. If the number of input variables are n, there are at most 2^n cases of possible maximum rules that have non-zero w_i, and the possible number of non-zero rules is eight for the case of $n = 3$. Possible rulse are smaller than 2^n when there are variables of premise which lies on midpoints of A_i. This fact coincides with the number of table terms which are used in MDI. If we use algebraic product for w_i,

$$w_i = A_{1i}(x_1^0)A_{2i}(x_2^0)A_{3i}(x_3^0). \quad (13)$$

The defuzzified ith value is

$$\begin{aligned} y_i^* &= \mu_{B_i^{-1}}(w_i) \\ &= \mu_{B_i^{-1}}(A_{i1}(x_1^0)A_{i2}(x_2^0)A_{i3}(x_3^0)). \end{aligned} \quad (14)$$

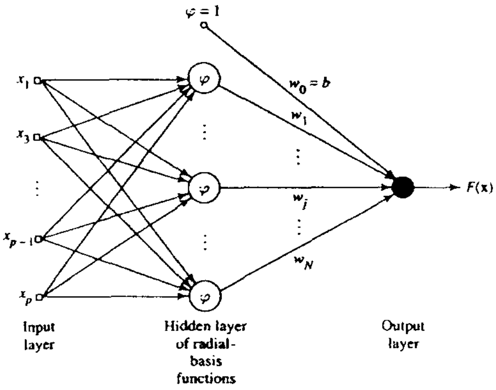
We define the overall defuzzified value y^* as Eq. (15) (Note that *weighed combination method* is usually used in Tsukamoto's defuzzification.),

$$y^* = \sum_{i=1}^{8} y_i^*. \quad (15)$$

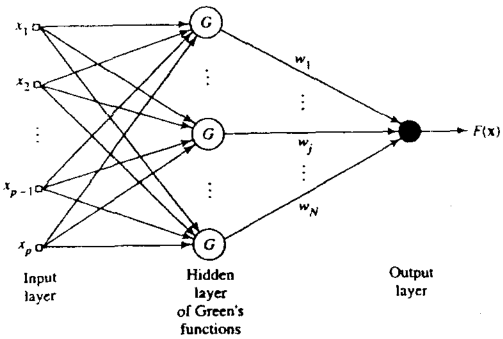
This results is equal to that of Eq. (4). We can easily verify that the cases of dimensions (one, two, four or more) in MDI produce the same results of previous special Tsukamoto's fuzzy reasoning in which the number of input variable is n.

4. Triangular Basis Function Networks

Typical RBF networks and regularization networks are shown in Fig. 3.



(a)



(b)

Fig. 3. (a) Radial basis function networks, (b) regularization networks. (From [11, p. 256, 260.)

4.1 RBF networks

RBF networks were originally proposed as an interpolation method, and their properties as interpolants have been extensively studied [8]. The radial basis function (RBF) technique consists of choosing a function F that has the following form [10];

$$F(X)=\sum_{i=1}^{N} w_i \varphi(\| X-C_i \|)+w_0 \quad (16)$$

where $\{ \varphi(\|X-C_i\|) | i = 1, 2, \ldots, N \}$ is a set of N arbitrary (generally nonlinear) functions, known as radial basis function, and $\| \ \|$ denotes a norm that is usually taken to be Euclidean. The known data points $C_i \in R^p$, $i = 1, 2, \ldots, N$ are taken to be the *centers* of the radial basis function. Theoretical investigations and practical results, however, seem to show that the type of nonlinearity $\varphi(\cdot)$ is not crucial to the performance of RBF networks [10].

Property 1 (Factorizable Radial Basis Function) : For a radial basis function φ we have

$$\varphi(\| X-C\|^2) = \varphi(|x_1-c_1|^2)\varphi(|x_2-c_2|^2)\ldots\varphi(|x_N-c_N|^2) \quad (17)$$

The synthesis of radial basis functions in many dimensions may be easier if they are factorizable. It can be easily proven that the only radial basis function which is factorizable is the Gaussian. A multidimensional Gaussian function can be represented as the product of lower dimensional Gaussians. Aside the implementation point of view, since it is difficult to imagine how neurons could compute $G(\|X-C_i\|^2)$ in a simple way for dimensions higher than two [11].

4.2 Regularization Networks

The principle of regularization is :

Find the function $F(X)$ that minimizes the cost functional $E(F)$, defined by

$$E(F) = E_S(F)+\lambda E_C(F) \quad (18)$$

where $E_S(F)$ is the *standard error term*, $E_C(F)$ is the *regularization term*, and λ is the *regularization parameter* [9, p. 247].

We may state that the solution to the regularization problem is given by the expansion

$$F(X)=\sum_{i=1}^{N} w_i G(X ; C_i) \quad (19)$$

where $G(X ; C_i)$ is the Green's function. For detail illustration of regularization problem and Green's function, see [9][11]. The RBF is a restricted version of the regularization function. The condition for this is *translational and rotational invariance.*

- *Translational invariance* : The Green's function $G(X ; C_i)$ centered at C_i will depend only on the *difference* between the argument X and C_i ; that is $G(X ; C_i) = G(X - C_i)$.
- *Translational and rotational invariance* : The Green's function $G(X ; C_i)$ centered at C_i will depend only on the *Euclidean norm* of X and C_i ; that is $G(X ; C_i) = G(\|X - C_i\|)$.

Under these conditions, the Green's function network must be a radial-basis function network as follow.

$$F(X)=\sum_{i=1}^{N} w_i G(\| X - C_i\|). \quad (20)$$

It is important, however, to realize that this solution differs from that of Eq. (16) in a fundamental respect: It is only when we set the regularization parameter λ equal to zero that the two solutions may become one and the same except w_0 [9].

4.3 Triangular basis function networks

Proposed TBF network is one kind of regularization networks. So the structure of TBF networks are equal to that of regularization networks.

Definition 1 (Triangular Basis Function) :

$$\Lambda(X ; C) \equiv \Lambda((X-C)_{\langle \tau 1, \tau 2\rangle}) = \prod_{K=1}^{P} ((X_K - C_K)_{\langle \tau 1K, \tau 2K\rangle}) \quad (21)$$

where

$$\Lambda(r_{\langle \tau 1, \tau 2\rangle}) = \begin{cases} \frac{r+\tau 1}{\tau 1}, & \text{for } -\tau 1 < r \le 0, \\ 1-\frac{r}{\tau 2}, & \text{for } 0 < r \le \tau 2, \\ 0, & \text{for otherwise.} \end{cases} \quad (22)$$

and P is the dimension of input space. See

Fig. 3 for graphical illustration. Then the TBF network is

$$
\begin{aligned}
F(X) &= \sum_{i=1}^{N} w_i \Lambda((X-C)_{\langle \tau 1, \tau 2\rangle}) \\
&= \sum_{i=1}^{N} w_i \left(\prod_{i=1}^{P} ((X_K - C_K)_{\langle \tau 1K, \tau 2K\rangle}) \right).
\end{aligned} \tag{23}
$$

Eqs (21) and (22) state the followings.

- Proposed TBF can be calculated only by *factorized form* if the input space is multidimensional.
- Proposed triangular basis function holds only the property of *translation invariance*.
- If the interval of each dimensional data of LUT is constant ($\tau 1 = \tau 2$; *rotational invariance*), triangular basis function becomes *radial basis function*.

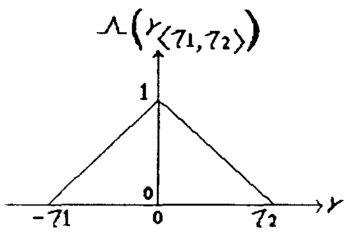


Fig. 4. Triangular basis function $\Lambda(r_{\langle \tau 1, \tau 2\rangle})$.

4. 4. Expression of multidimensional linear interpolation from triangular basis function networks

From Eq. (22)

$$
\Lambda((x-t)_{\langle \tau 1, \tau 2\rangle}) = \begin{cases} \dfrac{x-t+\tau 1}{\tau 1}, & \text{for } -\tau 1 < x-t \le 0, \\ 1-\dfrac{x-t}{\tau 2}, & \text{for } 0 < x-t \le \tau 2, \\ 0, & \text{for otherwise,} \end{cases} \tag{24}
$$

$$
= \begin{cases} \dfrac{x-(t-\tau 1)}{t-(t-\tau 1)}, & \text{for } -\tau 1 < x-t \le 0, \\ 1-\dfrac{x-t}{(t+\tau 2)-t}, & \text{for } 0 < x-t \le \tau 2, \\ 0, & \text{for otherwise.} \end{cases}
$$

We can easily verify that this is equal to $\{(u),(1-u)\}$ or $\{(v),(1-v)\}$ or $\{(w),(1-w)\}$ of Eq. (4). If we set w, C, $<\tau 1, \tau 2>$ to be value, position, and distances between C and nearby C, respectively, the output of TBF network is equal to Eq. (4) for three dimension, i.e., in Eq (20) w_i is corresponding to y_i, and $\{(u)$ or $(1-u)\}\{(v)$ or $(1-v)\}\{(w)$ or $(1-w)\}$ to $\Lambda((X-C)_{<\tau 1, \tau 2>})$ = $\prod_{K=1}$ to $_3((X_K-C_K)_{<\tau 1K, \tau 2K>})$ for three dimensions. We can also verify that the cases of n-dimension (one, two, four or more) in an MDI produce the same results of the corresponding TBF network.

5. Discussion

We showed two interesting results in this paper.

Firstly, multidimensional linear interpolation (MDI) is a special form of Tsukamoto's fuzzy reasoning. And if compatibility w_i is well defined, defuzzification strategy can be achieved simply by summing of each defuzzified ith value y_i^*. If we use the followings in Tsukamoto's method, the result is equal to an MDI.

① input variable : fuzzy singleton.
② membership system : STM system as discribed in section IV.
③ algebraic product for compatibility w_i.
④ final overall defuzzified value $y^* = \sum_{i=1}^{N} y_i^*$.

So, MDI is efficient than fuzzy reasoning because the former uses valid data whereas the later calculates all possible cases of rules even if they produce zero value. If we think input data are contaminated by noise, we can regard input value as fuzzy number when we use fuzzy reasoning method, but an MDI has no flexibility.

Secondly, MDI is a special form of regularization networks. If we use the followings in regularization networks, the result is equal to that of an MDI.

① Kernel in hidden layer of regularization networks : triangular basis function as discussed in section 3.3.
② w : value in an LUT.
③ C : position in an LUT.
④ $<\tau 1, \tau 2>$: distances between C and nearby C.

So, even if we can get the same output, an MDI is efficient than regularization networks because the former uses valid data whereas the later calculate all possible basis functions even if they produce zero value. So the MDI is efficient than regularization networks in the perspective of operation cost. But, in TBF networks we have flexibility of making nonlinear interpolated output simply by setting a new strategy for $<\tau 1,\ \tau 2>$ and w.

6. Conclusion

Firstly, we showed that an MDI is a special form of Tsukamoto's fuzzy reasoning. From this result, we found the overall defuzzification strategy can be accomplished by adding only each rule's defuzzified value if compatibility w_i is well defined. We compared both MDI and fuzzy reasoning.

Secondly, we showed that an MDI is a special form of regularization networks. For this purpose, we proposed a TBF network. Also we verified the condition when our proposed TBF becomes a well-known radial basis function. We compared both MDI and triangular basis function networks.

Further researches are necessary to find compatibility w_i which can be used in simple defuzzification strategies that the overall defuzzification can be accomplished by adding each defuzzified values as stated before. And there remains the problem of relationship among the MDI of tabulated function values at "random" points in n-dimensional space , fuzzy reasoning, and triangular basis function networks.

References

1. S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, 1996.

2. A. K. Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, 1989.

3. T. Sudkamp and R. J. Hammell II, "Interpolation, Completion, and Learning Fuzzy Rules," *IEEE Trans. on Syst. Man, and Cybern.*, vol. SMC-24, no. 2, pp. 332-342, Feb. 1994.

4. Y. Tsukamoto, "An Approach to Fuzzy Reasoning Method," in *Advances in Fuzzy Set Theory and Applications*, M. M. Gupta, R. K. Ragada, and R. R. Yager, Eds, Amsterdam : North-Holland, 1979.

5. K. S. Om and B. G. Min, "Relations between Multidimensional Linear Interpolation and Fuzzy Reasoning," *J. of Korea Fuzzy Logic and Intelligent Systems*, vol. 6, no. 4, pp. 34-38, 1996.

6. K. S. Om and B. G. Min, "Relations between Multidimensional Linear Interpolation and regularization networks," *J. of Korea Fuzzy Logic and Intelligent Systems*, vol. 7, no. 3, pp. 89-95, 1997.

7. W. H. Press, *Numerical Recipes in C*, Cambridge University Press, 1992.

8. M. J. D. Powell, "The Theory of Radial Basis Function Approximation in 1990," *Advances in Numerical Analysis*, vol. 3, W. Light, Ed. Oxford : Clarendon, pp. 105-210, 1992.

9. S. Haykin, *Neural Networks*, Macmillan, 1994.

10. M. J. D. Powell, "Radial Basis Function Approximation to Polynomials," *Numerical Analysis 1987 Proceedings*, pp. 223-241, Dundee, UK. 1988.

11. T. Poggio and F. Girosi, "Networks for Approximation and Learning," *Proceeding of the IEEE*, vol. 78, no. 9, pp. 1481-1497, Sept. 1990.