Relation between Multidimensional Linear Interpolation and Regularization Networks

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

*Dept. of Biomedical Engineering, College of Engineering, Seoul National University
†Dept. of Biomedical Engineering, College of Medicine, Seoul National University

Abstract

This paper examines the relation between multidimensional linear interpolation (MDI) and regularization networks, and shows that an MDI is a special form of regularization networks. For this purpose we propose a triangular basis function (TBF) network. Also we verified the condition when our proposed TBF becomes a well-known radial basis function (RBF).

I. Introduction

The training process of a neural network may be viewed as one of *curve fitting*[1, p. 855]. 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. One of applications of this method is the estimation of the pump output of artificial heart, and showed good performance [4]. Recently, Om *et al.* showed that MDI is a special form of Tsukamoto's fuzzy reasoning [5]. In the other view point, this paper examines the relation between MDI and regularization networks, and shows that an MDI is a special form of regularization networks. For this purpose we proposed triangular basis function networks. Also we verified when our proposed triangular basis function (TBF) becomes a well-known radial basis function (RBF).

This paper is organized as follows. We state an MDI and regularization networks in section II and III, respectively. In section IV, we derive the MDI from the proposed triangular basis function network. In section V, we summarize and discuss about our study. Finally, in section VI, conclusions are stated.

II. 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, and what we say LUT is rectangular data array throughout this paper. 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 [6].

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

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

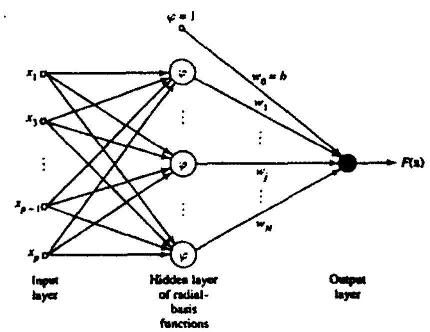
$$\begin{aligned} u &= \frac{x_1 - x_{1a}[m]}{x_{1a}[m+1] - x_{1a}[m]}, \\ v &= \frac{x_2 - x_{2a}[n]}{x_{2a}[n+1] - x_{2a}[n]}, \\ w &= \frac{x_3 - x_{3a}[r]}{x_{3a}[r+1] - x_{3a}[r]}. \end{aligned} \tag{5}$$

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

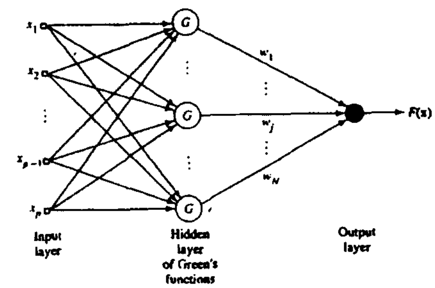
We can see the estimated y uses 2^n table terms if n-dimensions, and it satisfies 8 terms in the case of three dimensions as above.

III. Triangular Basis Function Network

In this section, we will state *radial basis function* (RBF) *networks*, *regularization networks*, and the proposed *triangular basis function* (TBF) *networks*. Typical RBF networks and regularization networks are shown in Fig. 1.



(a)



(b)

Fig. 1. (a) Radial basis function networks, (b) regularization networks. (From [11, p. 256, 260.)

3.1 RBF Networks

RBF networks were originally proposed as an interpolation method, and their properties as interpolants have been extensively studied [7]. It is now one of the main fields of research in numerical analysis. RBF networks have been shown to have universal approximation ability by Hartma *et al.* [8] and Park and Sandberg [9][10]. The radial basis function (RBF) technique consists of choosing a function F that has the following form [12];

$$F(X) = \sum_{i=1}^{N} w_i \varphi(\| X - C_i \|) + w_0 \quad (6)$$

where { $\varphi(\|X-C_i\|) | i$ = 1, 2, ... , 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, ..., 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 [12]. Some of $\varphi(\cdot)$ are listed in the followings [13][14][15][16].

1. Linear

$$\varphi(r) = r, \quad \text{for } r \geq 0. \quad (7)$$

2. Cubic

$$\varphi(r) = r^3, \quad \text{for } r \geq 0. \quad (8)$$

3. Thin-plate-spline function

$$\varphi(r) = \left(\frac{r}{\sigma}\right)^2 \ln\left(\frac{r}{\sigma}\right), \quad \text{for some } \sigma > 0, \text{ and } r \geq 0. \quad (9)$$

4. Gaussian function

$$\varphi(r) = \exp\left(-\frac{r^2}{2\sigma^2}\right), \quad \text{for some } \sigma > 0, \text{ and } r \geq 0. \quad (10)$$

5. Multiquadrics

$$\varphi(r) = \sqrt{r^2 + c^2}, \quad \text{for some } c > 0, \text{ and } r \geq 0. \quad (11)$$

6. Inverse multiquadrics

$$\varphi(r) = \frac{1}{\sqrt{r^2 + c^2}}, \quad \text{for some } c > 0, \text{ and } r \geq 0. \quad (12)$$

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

$$\varphi(\| X - C \|^2) = \varphi(|x_1 - c_1|^2)\varphi(|x_2 - c_2|^2)\ldots\varphi(|x_N - c_N|^2) \quad (13)$$

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 [17].

3.2 Regularization Networks

The principle of regularization is :

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

$$E(F) = E_S(F) + \lambda E_C(F) \quad (14)$$

where $E_S(F)$ is the *standard error term*, $E_C(F)$ is the *regularization term*, and λ is the *regularization parameter* [11, p. 247].

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

$$F(X) = \sum_{i=1}^{N} w_i G(X\,;\,C_i) \quad (15)$$

where $G(X\,;\,C_i)$ is the Green's function. For detail illustration of regularization problem and Green's function, see [11][17]. 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.

$$F(X) = \sum_{i=1}^{N} w_i G(\| X - C_i \|). \qquad (16)$$

It is important, however, to realize that this solution differs from that of Eq. (6) in a fundamental respect : The solution of definition given in Eq. (16) for the weight vector w. 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 [11].

3.3 Triangular Basis Function Networks

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

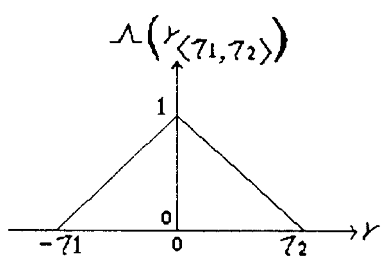


Fig. 3. Triangular basis function $\Lambda(r_{<\tau 1,\tau 2>})$.

Definition 1 (Triangular Basis Function) :

$$\Lambda(X\ ;C) \equiv \Lambda((X-C)_{<\tau 1,\tau 2>}) = \prod_{K=1}^{P}((X_K - C_K)_{<\tau 1K,\tau 2K>}) \qquad (17)$$

where

$$\Lambda(r_{<\tau 1,\tau 2>}) = \begin{cases} \dfrac{r+\tau 1}{\tau 1}, & \text{for } -\tau 1 < r \le 0, \\ 1-\dfrac{r}{\tau 2}, & \text{for } 0 < r \le \tau 2, \\ 0, & \text{for otherwise.} \end{cases} \qquad (18)$$

and P is the dimension of input space. See Fig. 3 for graphical illustration. Then the TBF network is

$$\begin{aligned} F(X) &= \sum_{i=1}^{N} w_i \Lambda((X-C)_{<\tau 1,\tau 2>}) \\ &= \sum_{i=1}^{N} w_i \Big(\prod_{i=1}^{P}((X_K - C_K)_{<\tau 1K,..\tau 2K>} \end{aligned} \qquad (19)$$

Eqs (17) and (18) 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.*

IV. Expression of Multidimensional Linear Interpolation from Triangular Basis Function Network

From Eq. (18)

$$\Lambda((x-t)_{<\tau 1,\tau 2>})$$

$$= \begin{cases} \dfrac{x-t+\tau 1}{\tau 1}, & \text{for } -\tau 1 < x-t \le 0, \\ 1-\dfrac{x-t}{\tau 2}, & \text{for } 0 < x-t \le \tau 2, \\ 0, & \text{for otherwise,} \end{cases} \qquad (20)$$

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

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 $\Lambda((X-C)_{<\tau1,\tau2>}) = \prod_{K=1 \text{ to } 3}((X_K-C_K)_{<\tau1K,\tau2K>})$ 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.

V. Discussion

We showed an interesting result in this paper, which multidimensional linear interpolation (MDI) is a special form of regularization networks. If we use the followings in regularization network, the result is equal to that of an MDI.

① Kernel in hidden layer of regularization network : triangular basis function as discussed in section 3.3.
② w : value in an LUT.
③ C : position in an LUT.
④ $<\tau1, \tau2>$: distances between C and nearby C.

At this point, we need to compare both methods. 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 even if we can get the same output, the MDI is efficient than regularization network in the perspective of operation cost. But, in TBF network we have flexibility of making nonlinear interpolated output simply by setting a new strategy for $<\tau1, \tau2>$ and w.

VI. Conclusion

It is known that multidimensional linear interpolation is a special form of Tsukamoto's fuzzy reasoning [5]. In the other view point, we showed that an MDI is a special form of regularization networks in this paper. For this purpose we proposed a triangular basis function (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 network in section V. Further researches are necessary to find the relation between MDI of tabulated function values at 'random' points in n-dimensional space and triangular basis function network.

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] K. S. Om, W. W. Choi, J. M. Ahn, S. K. Park, Y. H. Jo, J. S. Choi, J. J. Lee, H. C. Kim, and B. G. Min, "Experimental Approach for the Estimation of Cardiac Output of Left Ventricular Assist Device Using Multi-dimensional Interpolation Technique," *KOSOMBE Spring Conference*, pp. 232-234, 1996.

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

[6] W. H. Press, *Numerical Recipes in C*, Cambridge University Press, 1992.

[7] 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.

[8] E. J. Hartman, J. D. Keeler, and J. M. Kowalski, "Layered Neural Networks with Gaussian Hidden Units as Universal Approximation," *Neural Computa.*, vol. 2, pp. 210-215, 1990.

[9] J. Park and I. W. Sandberg, "Universal Approximation Using Radial Basis Function Networks," *Neural Computa.*, vol. 3, pp. 246-257, 1991.

[10] J. Park and I. W. Sandberg, "Approximation and Radial Basis Function Networks," *Neural Computa.*, vol. 5, pp. 305-316, 1993.

[11] S. Haykin, *Neural Networks*, Macmillan, 1994.

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

[13] B. A. Whitehead, T. D. Choate, "Cooperative-Competitive Genetic Evolution of Radial Basis Function Centers and Widths for Time Series Prediction," *IEEE Trans. Neural Networks*, vol. 7, no. 4, July, 1996.

[14] D. S. Broomhead and D. Lowe, "Multivariable Function Interpolation and Adaptive Networks," *Complex Syst.*, vol. 2, pp. 321-355, 1988.

[15] M. T. Musavi, W. Ahmed, K. H. Chan, K. B. Faris, and D. M. Hummels, "On the Training of Radial Basis Function Classifier," *Neural Networks*, vol. 5, pp. 595-603, 1992.

[16] S. Chen, C. F. N. Cowan, and P. M. Grant, "Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks," *IEEE Trans. Neural Networks*, vol. 2, no. 2, pp. 302-309, 1991.

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