A fast learning method using cost matrices for morphological neural networks

***Hajime Nobuhara, **Barnabas Bede, and *Kaoru Hirota**

*Department of Computational Intelligence and Systems Science,

Tokyo Institute of Technology, G3-49, 4259 Nagatsuta, Midiri-ku,

Yokohama 226-8502, Japan

E-mail: {nobuhara, hirota} @hrt.dis.titech.ac.jp

**Department of Mechanical and System Engineering, Budapest Tech,

Nepszinhaz, u.8, H-1081 Budapest, Hungary

E-mail: bede.barna@bgk.bmf.hu

Abstract: A fast learning method for morphological neural networks which are formulated based on max plus algebra, is proposed. The proposed fast learning method is deduced from the idempotent properties of max plus algebra and cost matrices. Through experiments using artificial training data sets, it is conformed that the computation time of the proposed method is decreased into 60.9-68.6% and 40.9 – 48.3% of that of the conventional one, under the condition that the number of middle layer nodes is 4 and 8, respectively.

Keywords: max plus algebra, neural networks, gradient method, mathematical morphology

1 Introduction

The recent resurgence of interest in artificial neural networks has brought a deluge of publications to the field. Application of neural networks are appearing in a wider variety of fields each year. The neural networks are formulated based on not only ordinal algebra but also other algebra, e.g., fuzzy algebra [1-3], and max plus algebra[4][5]. The max plus algebra based neural networks are generally called morphological neural networks which can treat non-linear phenomena in the ordinal algebra as linear ones. In this paper, a fast learning method for the morphological neural networks is proposed based on the idempotent properties of max plus algebra and two cost matrices.

An experimental comparison using artificial training data sets is performed and the effectiveness of the proposed fast learning method is confirmed.

2 Morphological Neural Networks

2.1 Networks Representation by Morphological Operations

Morphological neural networks are constructed based on morphological operators [4][5] which are defined as

$$\max\left\{a, b\right\}, \tag{1}$$

$$a + b, \tag{2}$$

instead of standard addition and multiplication in ordinal algebra, respectively. This paper considers morphological neural networks with three layers (input, middle, and output layers), as shown in Fig. 1, where the input, middle, and output layer vectors with N_I, N_m, and N_o dimensions respectively, are denoted by

$$\mathbf{x} = \left\{x_1, x_2, \ldots, x_{N_I}\right\}, \quad \left(\in \mathbf{R}^{N_I}\right) \tag{3}$$

$$\mathbf{m} = \left\{m_1, m_2, \ldots, m_{N_m}\right\}, \quad \left(\in \mathbf{R}^{N_m}\right) \tag{4}$$

$$\tilde{\mathbf{y}} = \left\{\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_{N_o}\right\}, \quad \left(\in \mathbf{R}^{N_o}\right) \tag{5}$$

and the connections (weights coefficients) between the input and middle layer are defined as

$$w_{jl}\left(\in \mathbf{R}\right) \quad j = 1, 2, \ldots, N_I, l = 1, 2, \ldots, N_m, \tag{6}$$

and the connection weights between the middle and output layer are defined as

$$w'_{pq}\left(\in \mathbf{R}\right) \quad p = 1, 2, \ldots, N_m, q = l = 1, 2, \ldots, N_o. \tag{7}$$

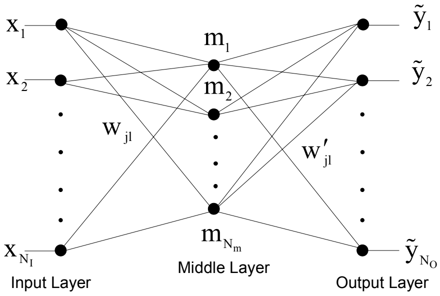

Figure 1: Topology of morphological neural networks

According to the definition of morphological operations Eq. (1) and (2), the middle layer vector can be calculated as

$$m_i = \max_{j=1}^{N_I} \left\{ w_{ji} + x_j \right\}, \quad i = 1, 2, \ldots, N_m , \tag{8}$$

and the output layer vector is calculated as

$$\tilde{y}_q = \max_{p=1}^{N_m} \left\{ w'_{pq} + m_p \right\}, \quad q = 1, 2, \ldots, N_o . \tag{9}$$

From Eqs. (8) and (9),

$$\tilde{y}_q = \max_{p=1}^{N_m} \left\{ w'_{pq} + \max_{j=1}^{N_I} \left\{ w_{jp} + x_j \right\} \right\}, \quad q = 1, 2, \ldots, N_o , \tag{10}$$

is obtained.

2.2 A Learning Method for Morphological Neural Networks

To train the morphological neural networks, the train data sets

$$\left(\mathbf{x}^{(k)}, \mathbf{y}^{(k)} \right), \quad k = 1, 2, \ldots, D_n , \tag{11}$$

are considered. The output vector $\tilde{\mathbf{y}}^{(k)}$ is obtained by the morphological neural networks with respect to $\mathbf{x}^{(k)}$ based on Eq. (10), and the difference between $\tilde{\mathbf{y}}^{(k)}$ and $\mathbf{y}^{(k)}$ is defined by

$$F\left(\mathbf{y}^{(k)}, \tilde{\mathbf{y}}^{(k)} \right) = \sum_{q=1}^{N_o} \left(y_q^{(k)} - \tilde{y}_q^{(k)} \right)^2 . \tag{12}$$

By using the difference function Eq. (12), the updating the weight of the morphological neural networks is performed according to 2.2.1 and 2.2.2.

2.2.1 Updating of the weight between the output and middle layer

The updating of the weight is based on the gradient method [2][3] as

$$w'_{pq} = w'_{pq} - \alpha \frac{\partial F\left(\mathbf{y}^{(k)}, \tilde{\mathbf{y}}^{(k)}\right)}{\partial w'_{pq}}, \quad p = 1,2,\ldots,N_m, \quad q = 1,2,\ldots,N_O , \tag{13}$$

where

$$\begin{aligned} \frac{\partial F\left(\mathbf{y}^{(k)}, \tilde{\mathbf{y}}^{(k)}\right)}{\partial w'_{pq}} &= 2 \cdot \left(y_q^{(k)} - \tilde{y}_q^{(k)}\right) \\ &\cdot \frac{\partial}{\partial w'_{pq}} \left\{ w'_{pq} + \max_{j=1}^{N_I} \left\{ w_{jp} + x_j^{(k)} \right\}, \quad \max_{\substack{r=1 \\ r \neq p}}^{N_m} \left\{ w'_{rq} + \max_{j=1}^{N_I} \left\{ w_{jr} + x_j^{(k)} \right\} \right\} \right\}. \end{aligned} \tag{14}$$

The derivation term of Eq. (14) is defined as

$$\begin{aligned} &\frac{\partial}{\partial w'_{pq}} \left\{ w'_{pq} + \max_{j=1}^{N_I} \left\{ w_{jp} + x_j^{(k)} \right\}, \quad \max_{\substack{r=1 \\ r \neq p}}^{N_m} \left\{ w'_{rq} + \max_{j=1}^{N_I} \left\{ w_{jr} + x_j^{(k)} \right\} \right\} \right\} \\ &= \begin{cases} 1 & \text{if} \quad w'_{pq} + \max_{j=1}^{N_I} \left\{ w_{jp} + x_j^{(k)} \right\} \geq \max_{\substack{r=1 \\ r \neq p}}^{N_m} \left\{ w'_{rq} + \max_{j=1}^{N_I} \left\{ w_{jr} + x_j^{(k)} \right\} \right\}, \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{15}$$

2.2.2 Updating of the weight between the middle and input layer

If $w_{jm} + x_j^{(k)} \geq \max_{\substack{l=1 \\ l \neq j}}^{N_I} \left\{ w_{lm} + x_l^{(k)} \right\}$, the weight coefficient w_{jm} between the middle and input layer is updated as

$$w_{jm} = w_{jm} - \beta \sum_{i=1}^{N_o} \delta(m,i) \tag{16}$$

$$\delta(m,i) = \begin{cases} y_i^{(k)} - \tilde{y}_i^{(k)} & \text{if} \quad w'_{mi} + \max_{l=1}^{N_I} \left\{ w_{lm} + x_l^{(k)} \right\} \geq \max_{\substack{r=1 \\ r \neq p}}^{N_m} \left\{ w'_{ri} + \max_{l=1}^{N_I} \left\{ w_{lr} + x_l^{(k)} \right\} \right\}, \\ 0 & \text{otherwise.} \end{cases} \tag{17}$$

3 A Fast Learning Method for Morphological Neural Networks Based on Cost Matrices

3.1 Definition of Cost Matrices

In order to formulate the proposed fast learning method for morphological neural networks, two cost matrices are defined as

$$C_1(j,m)=\begin{cases}1 \quad \text{if} \quad w_{jm}+x_j^{(k)} \geq \max\limits_{\substack{l=1 \\ l\neq j}}^{N_I}\left\{w_{lm}+x_l^{(k)}\right\}, \\ \qquad 0 \qquad \text{otherwise,}\end{cases} \tag{18}$$

and

$$C_2(p,q)=\begin{cases}y_q^{(k)}-\tilde{y}_q^{(k)} \quad \text{if} \quad w'_{pq}+\max\limits_{j=1}^{N_I}\left\{w_{jp}+x_j^{(k)}\right\} \geq \max\limits_{\substack{r=1 \\ r\neq p}}^{N_m}\left\{w'_{rq}+\max\limits_{j=1}^{N_I}\left\{w_{jr}+x_j^{(k)}\right\}\right\}, \\ \qquad 0 \qquad \text{otherwise.}\end{cases} \tag{19}$$

3.2 A Fast Learning Method Procedure based on Cost Matrices

Based on the definition of cost matrices (Eqs. (19) and (20)), the proposed fast learning method procedure for morphological neural networks can be formulated as follows.

Step. 1: Initialize the weight coefficients w_{jm} and w'_{mn}, $j=1,2,\ldots,N_I$, $m=1,2,\ldots,N_m$, and $n=1,2,\ldots,N_O$ randomly.

Step. 2: With respect to the training data set $\left(\mathbf{x}^{(k)},\mathbf{y}^{(k)}\right)$, the output $\tilde{\mathbf{y}}^{(k)}$ is calculated by Eq. (10), and the cost matrices C_1 and C_2 are also simultaneously calculated through the calculation process (Eq. (10)).

Step. 3: By using the cost matrices, the updating process (Eqs. (13) – (15)) of weight coefficients between the output and middle layer can be modified as:

$$w'_{pq}=w'_{pq}-\alpha\frac{\partial F\left(\mathbf{y}^{(k)},\tilde{\mathbf{y}}^{(k)}\right)}{\partial w'_{pq}}, \quad p=1,2,\ldots,N_m, \quad q=1,2,\ldots,N_O, \tag{20}$$

$$\frac{\partial F\left(\mathbf{y}^{(k)},\tilde{\mathbf{y}}^{(k)}\right)}{\partial w'_{pq}} = 2\cdot\left(y_q^{(k)} - \tilde{y}_q^{(k)}\right)$$
$$\cdot\frac{\partial}{\partial w'_{pq}}\left\{w'_{pq} + \max_{j=1}^{N_I}\left\{w_{jp} + x_j^{(k)}\right\},\quad \max_{\substack{r=1\\ r\neq p}}^{N_m}\left\{w'_{rq} + \max_{j=1}^{N_I}\left\{w_{jr} + x_j^{(k)}\right\}\right\}\right\}, \tag{21}$$

and

$$\frac{\partial}{\partial w'_{pq}}\left\{w'_{pq} + \max_{j=1}^{N_I}\left\{w_{jp} + x_j^{(k)}\right\},\quad \max_{\substack{r=1\\ r\neq p}}^{N_m}\left\{w'_{rq} + \max_{j=1}^{N_I}\left\{w_{jr} + x_j^{(k)}\right\}\right\}\right\}$$
$$= \begin{cases} 1 & \text{if } \quad C_2\left(p,q\right)\neq 0, \\ 0 & \quad\text{otherwise.}\end{cases} \tag{22}$$

Step. 4: The updating process (Eqs. (16) and (17)) of weight coefficients $w_{jm}, j = 1,2,\ldots,N_O, m = 1,2,\ldots,N_m$, between the middle and input layer can be modified as follows:

If $C_1\left(j,m\right) = 1$ then

$$w_{jm} = w_{jm} - \beta\sum_{i=1}^{N_o} C_2\left(m,i\right). \tag{23}$$

Step. 5: If a termination condition

$$\sum_k \left|\tilde{\mathbf{y}}^{(k)} - \mathbf{y}^{(k)}\right| < \varepsilon, \tag{24}$$

is satisfied, then the learning process is stopped, otherwise go to Step 2.

4 Experimental Comparisons

In order to confirm the effectiveness of the proposed method, an experimental comparison using artificial training data sets is performed. In this experiment, the conventional method and proposed one correspond to the learning algorithm described in section 2 and 3, respecrively, and the computation time of them are measured. The artifical training data sets from 100 to 1,000 are randomly generated and the morphological neural networks with 4 and 8 layers are trained by the proposed method and conventional one, respectively. Figures 2 and 3 show the computation time comparision of the proposed and conventional method with respect to the number of data sets, under the condition that the middle layer is 4 and 8 nodes, respectively. In Figs. 2 and 3, it is confirmed that the computational time of the proposed method is decreased into 60.9-68.6% and 40.9 – 48.3% of that of the conventional one, under the condition that the number of middle layer nodes is 4 and 8, respectively.

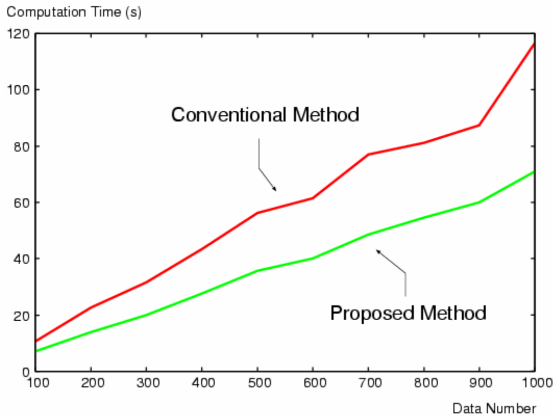

Figure 1. Computation time comparision of the proposed method and conventional one with respect to data number (middle layer = 4 nodes)

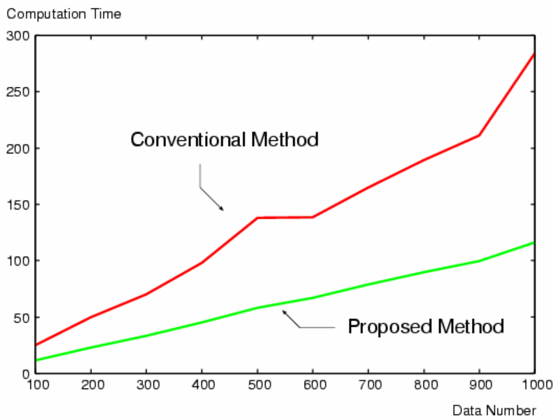

Figure 2. Computation time comparision of the proposed method and conventional one with respect to data number (middle layer = 8 nodes)

5 Conclusions

A fast learning method for morphological neural networks which is based on max plus algebra is proposed. The max plus algebra employs max and standard addition instead of standard addition and multiplication, respectively, and the proposed learning method is deduced from the properties of idempotent operation of max plus algebra and cost matrices.

In order to confirm the effectiveness of the proposed method, an experimental comparison using artificial training data (100 – 1,000) sets is performed. In this experiment, it is conformed that the computation time of the proposed method is decreased into 60.9-68.6% and 40.9 – 48.3% of that of the conventional one, under the condition that the number of middle layer nodes is 4 and 8, respectively.

References

[1] B. Gabrys and A. Bargiela: General Fuzzy Min-Max Neural Network for Clustering and Classification, IEEE Transaction on Neural Networks, Vol. 11, No. 3, May, 2000, pp. 769 – 783

[2] W. Pedrycz: Neurocomputations in Relational Systems, IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 13, No. 3, March, 1991, pp. 289 – 297

[3] J. V. Oliveira: Neuron inspired learning rules for fuzzy relational structures, Fuzzy Sets and Systems, Vol. 57, No. 1, 1993, pp. 41 – 53

[4] J. L. Davidson and G. X. Ritter: A Theory of Morphological Neural Networks, SPIE, Vol. 1215, 1990, pp. 378 – 388

[5] G. X. Ritter, D. Li and J. N. Wilson: Image Algebra and its Relationship to Neural Networks, SPIE, Vol. 1098, 1989, pp. 90 – 101