2010 International Symposium on Nonlinear Theory and its Applications
NOLTA2010, Krakow, Poland, September 5-8, 2010

Effect of Piecewise Linear Function on Maximum-Flow Neural Network

Masatoshi Sato†, Hisashi Aomori‡, and Mamoru Tanaka†

†Department of Electrical and Electronics Engineering ,Sophia University
7-1, Kioi-cho, Chiyoda-ku, Tokyo 102-8554, Japan
Phone:+81-3-3238-3878, Fax:+81-3-3238-3321
Email: masatoshi@tlab.ee.sophia.ac.jp
‡Department of Electrical Engineering Tokyo University of Science
2641 Yamazaki, Noda-shi, Chiba, 278–8510 Japan
Phone : +81–4–7124–1501 (Ext. 3722), Fax: +81–4–7124–9367
Email : aomori@ieee.org

Abstract—In our previous research, the Maximum-Flow Neural Network (MF-NN) was proposed, and we showed that the MF-NN is possible to solve any maximum-flow problems. Each neuron of the MF-NN is connected by nonlinear resistances with the saturation saturation property. However, the conventional MF-NN using the sigmoidal function has problems where the sigmoidal function dose not converge to $0, 1$ that is saturation value. In this research, the saturation property is improved by using the picewise linear function. Moreover, this novel method is possible to considerably reduce a calculation cost.

1. Introduction

The spread of the Internet continues to increase in modern society. Thus, having a method to send data quickly with a little loss is very important. This problem is commonly known as the maximum flow problem [1, 2], and the maximum flow algorithm offers the best solution to the problem of determining which route is appropriated to exchange data. Hence, the importance of the maximum flow algorithm is growing from the viewpoint of transportation capacity. The maximum flow problem involves streaming a large quantity of the flow from a starting point to a terminal point in the given network. In general, an information stream in a branch that satisfies the following conditions is called a flow; the flow does not exceed the capacity of the branch (the capacity condition), and the sum of the inflows is equal to the sum of the outflows (the flow preservation condition) on each node. The nodes incident to a branch exist in three-dimensional spaces in a network, and they are connected to the branch such that a graph consists of nodes and branches. The capacity of the branch (communication capacity) and the gain exist in each branch. The capacity indicates the limit of inflow or outflow, and the gain can enlarge or reduce the flow [3].

Over the years, various algorithms have been developed to solve the maximum flow problem. The Ford-Fulkerson algorithm [4, 5] and the preflow-push algorithm [6, 7] are two well known methods developed to solve this maximum flow problem. However, these algorithms are for computation by sequential machines without analog parallelism.

In our previous research, we proposed the Maximum-Flow Neural Network (MF-NN) [8], and we showed that the MF-NN is possible to solve any maximum-flow problems. The MF-NN in which each branch nonlinearity has a saturation characteristic and by which the maximum flow problem can be solved by using analog high-speed parallel processing [9, 10]. In a nonlinear resistive network, each nonlinear resistance with the saturation for the $I-V$ characteristic is described by the sigmoidal function. Originally, to fill the maximum flow algorithm, it is necessary to use the monotone increasing nonlinear function $f(x)$ of which $0 \leqq f(x) \leqq 1$ consists. Since nonlinear function $f(x)$ that uses the sigmoidal function is defined by $0 < f(x) < 1$, the function is approximately converged in the vicinity of $f(x) = 0, f(x) = 1$. However, the sigmoidal function has the character that the differential coefficient becomes very small around the convergence value. Therefore, there is a problem that takes an awful lot of convergence time of the solution of the state equation very much.

In this paper, we propose a novel MF-NN using the piecewise linear (PWL) function for solving this problem. Additionally, we show that the computation time can be greatly decreased. The relativity of the convergence time and the accuracy of the solution by the difference of the gain of the sigmoidal function is simulated. And, the convergence time and the solution are compared by MF-NN using the PWL function, and results are discussed.

2. The maximum flow algorithm

The maximum flow theorem is one of the most basic theorems of the network flow problem. When a positive constant branch capacity c_k is allocated in each branch $b_k \in B(N)$ on network N as a graph, the N is a communication network or transportation network. It is shown as $N = [V(N), B(N)]$ for nodes set $V[N]$ and branches set $B[N]$. The notation $C(N)$ is a set of branch capacity $\{c_k\}$. A directed branch b_k connected from node v_i to v_j on N is denoted by $b_k = (v_i, v_j) = b_{ij}$. A flow f from $s \in V(N)$ to $t(t \neq s) \in V(N)$ in the communication network N is defined by

$$\sum_{v_j \in \Gamma(v_i)} f(v_i, v_j) - \sum_{v_j \in \Gamma^{-1}(v_i)} f(v_j, v_i) = \begin{cases} F : v_i = s \\ -F : v_i = t \\ 0 : v_i \neq s, t, \end{cases} \quad (1)$$

$$0 \leqq f\left(v_i, v_j\right) \leqq c\left(v_i, v_j\right),$$
$$\left(v_i, v_j\right) \in B(N), \quad (2)$$

where

$$\Gamma(v_i) = v_j \mid (v_i, v_j) \in B(N),$$
$$\Gamma^{-1}(v_i) = v_j \mid (v_j, v_i) \in B(N). \quad (3)$$

$F = F(f)$ in Eq. (1) is the value of flow f, and node s and t are the source and the sink respectively.

Let the left part of Eq. (1) be the flow that flows out from v_i, then the Eq. (1) represents the restriction, where the flow from source s is F and the flow from sink t is $-F$ and the flow from arbitrary node $v_i \neq s, t$ is 0. Also, if the flow $f(v_i, v_j)$ that flows in each $(v_i, v_j) \in B(N)$ is the branch flow, then Eq. (2) represents the restriction where the branch flowing on each branch b_{ij} flows only in the direction from v_i to v_j and it does not exceed the branch capacity $c(v_i, v_j)$. In the communication network N, a branch set (X, Y) is defined as

$$(X, Y) = \{(v_i, v_j) \in B(N) | v_i \in X, v_j \in Y\}, \quad (4)$$

where $X, Y \subset V(N)$. The branch class (X, Y) has a source on X and sink on Y. For arbitrary flow f, the flow $f(X, Y)$ which flows (X, Y) is given by

$$f(X, Y) = \sum_{(v_i, v_j) \in (X,Y)} f(v_i, v_j). \quad (5)$$

In the flow f of the communication network N, a flow f_0 that gives the maximum value of $F(f)$ expressed by $F_0 = \max[F(f)]$ is called the maximum-flow.

3. The maximum-flow neural network

3.1. Conventional MF-NN

The network topology of the MF-NN is shown in Fig. 1. An input layer and an output layer of the network are corresponding to the start point S and the terminal point T, and each point corresponds to a single neuron, and layer number of inner layer is m. Additionally, the propagation between the neighboring neurons is interactive. That is, the MF-NN has feedback connections. The layer number of inner layer changes depending on how to connect the neurons. The structure is determined by a given transportation network. Fig. 2 shows the connection between the neuron v_i and the neuron v_j by the nonlinear resistive network. The A_{ij} is nonlinear resistance that exists between the neuron v_j and the neuron v_i. The MF-NN has the saturation characteristic such that the entire network converges to the equilibrium state if a certain amount of the current in the starting point s goes out. The $I - V$ branch characteristic from the neuron v_i to the neuron v_j is described by

$$I_{ij} = A_{ij} f(u_i - u_j), \quad (6)$$

where

$$f(x) = \frac{1}{1 + exp(-ax)}, \quad (7)$$

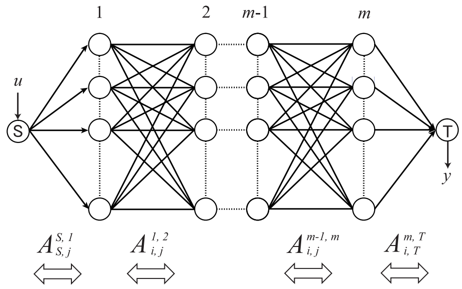

Figure 1: The network topology of MF-NN

A_{ij} is the maximum capacity $c(v_i, v_j)$ of the current in b_{ij} and a positive constant value. I_{ij} is a current that flows from the neuron v_i to v_j, and u_i and u_j are node voltages of the neuron v_i and v_j respectively. The constant a is the gain. The state equation with respect to the neuron v_i is given by

$$\begin{aligned} F_i(u_i) &\equiv C_i \frac{du_i}{dt} \\ &= -\sum_{v_j \in \Gamma(v_i)}^{n} A_{ij} f(u_i - u_j) \\ &\quad + \sum_{v_k \in \Gamma^{-1}(v_i)}^{n} A_{ki} f(u_k - u_i), \end{aligned} \quad (8)$$

where

$$\Gamma(v_i) = v_j | (v_i, v_j) \in B(N),$$
$$\Gamma^{-1}(v_i) = v_k | (v_k, v_i) \in B(N). \quad (9)$$

C_i is a capacitor that exists between the neuron v_i and the ground. By solving the differential equation Eq. (8) concerning the neuron v_i ($i = 1, 2, \cdots, n$), the state of each neuron (node voltage) u_i can be obtained. As a result, the potential differences between neurons and the branch current value corresponding to the voltage differences between each neuron are obtained.

Combining the equations of the maximum flow algorithm were shown from Eq. (1) to Eq. (4), the state equa-

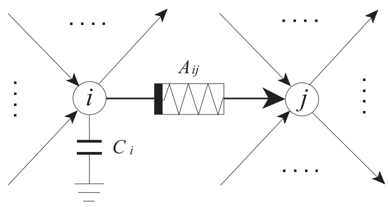

Figure 2: Association between neuron i and j

tion of the MF-NN, Eq. (8) can be rewritten as

$$\begin{aligned}\lim_{t\to\infty} C_i \frac{du_i}{dt} &= -\sum_{v_j\in\Gamma(v_i)}^{n} A_{ij} f(u_i - u_j) \\ &\quad + \sum_{v_k\in\Gamma^{-1}(v_i)}^{n} A_{ki} f(u_k - u_i) \\ &= \begin{cases} -F : v_i = s \\ F : v_i = t \\ 0 : v_i \neq s, t, \end{cases} \end{aligned} \tag{10}$$

$$0 \leqq A_{ij} f(u_i - u_j) \leqq A_{ij} = c\left(v_i, v_j\right), \quad \left(v_i, v_j\right) \in B(N). \tag{11}$$

Since the equilibrium state to fill Eq. (10) is described by

$$\lim_{t\to\infty} C_i \frac{du_i}{dt} = 0 \ \ (v_i \neq s, t), \tag{12}$$

Eq. (12) shows the state where the network is saturated. That is, only when the MF-NN is saturated, the state equation conform to the maximum flow algorithm. Moreover, since each current value (the flow) I_{ij} becomes the maximum value in the saturated state, the F of Eq. (10) shows the maximum flow F_0.

3.2. Problems of Conventional MF-NN

Essentially the nonlinear function $f(x)$ should be used under the condition that $f(x)$ satisfies the condition $0 \leqq I_{ij} \leqq A_{ij}$. However, the $I - V$ characteristic of the conventional MF-NN depends on the sigmoidal function as shown in Eq. (6), (7). The sigmoidal function is defined as $0 < f(x) < 1$, therefore $f(x)$ is approximately converged in the vicinity of $f(x) = 0, f(x) = 1$.

In conventional MF-NN, the condition $0 \leqq I_{ij} \leqq A_{ij}$ is realized by raising the gain a. For example, the velocity that approaches $f(x) = 0, 1$ can be changed by the difference of the gain a, and $f(x)$ is converged early by raising the gain a. However, since differential coefficient approaches 0, it takes a lot of computing time until convergent solutions of the simultaneous differential equation of MF-NN are obtained.

3.3. Novel MF-NN using PWL function

To solve the problem of conventional MF-NN, other nonlinear function $f(x)$ is used. Ideal $f(x)$ to fill the maximum flow algorithm is given by

$$f(x) = \begin{cases} 1 \ \ (b \leqq x), \\ g(x) \ \ (0 \leqq x < b), \\ 0 \ \ (x < 0). \end{cases} \tag{13}$$

where, b is positive constant and $g(x)$ is monotonically increasing function. Since $f(x)$ is constantly a positive function, the backflow phenomenon where a positive current flowed to a negative potential difference was occasionally generated in conventional MF-NN. To avoid this backflow phenomenon, the function that passes point $f(0) = 0$ and $f(a) = 1$ is selected.The nonlinear function $f(x)$ that contains those conditions is defined by,

$$f(x) = \begin{cases} 1 \ \ (b \leqq x), \\ x \ \ (0 \leqq x < b), \\ 0 \ \ (x < 0). \end{cases} \tag{14}$$

Since the PWL function has the feature with constant differential coefficient, the function has an advantage that the solution can be obtained at the very high speed in the numerical analysis of the nonlinear simultaneous differential equation that uses Newton method.

4. Simulation Results

In this research, we use a network which has mutual coupling as shown in Fig. 3. This network has the start-point node u_s, the inner layer nodes u_1, u_2, u_3, u_4, and the terminal node u_t. The initial voltage of the network analyzed is set to $u_s = 10V$. And, there are mutual coupling branches $b_{i,j}(i, j = 1, 2, 3, 4)$, and the branch capacities $c_{i,j}$ are given on each branch $b_{i,j}$ as shown in Fig. 3. Since $c_{i,j} \neq c_{j,i}$, we show that the novel MF-NN can solve complex problems in maximum flow problems.

The problem of the conventional MF-NN is that the error of a maximum flow occurs especially in the case of mutual coupling network analysis. The network shown in Fig. 3 is small-scale, but enough results to compare the performances is obtained.

The influence on the solution by the difference in gain a of various patterns is entertained. The content b of the PWL function Eq. (14) is set to $b = 1$. Under these conditions, the simulation results are compared.

Table 1 is a table where the current value (flow) of each branch by the difference of each gain a and each maximum flow are shown. Table 2 shows the calculation frequency where the node voltage of node 1 converges. As a result, in the case of conventional MF-NN, the error margin of the maximum flow becomes small when the the gain a is raised. However, there is a fault of an awful lot the calculation frequency, and taking a lot of converging time. On the other hand, proposed MF-NN using PWL function doesn't have the error margin of the maximum flow, and the calculation processing time is also very fast compared with conventional MF-NN.

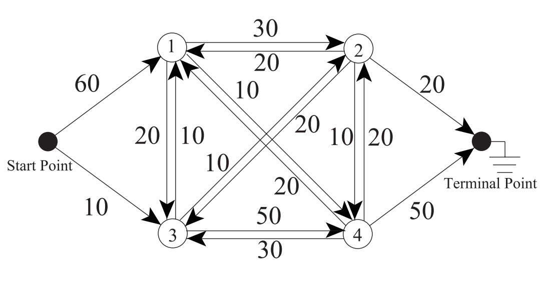

Figure 3: Analyzed network

Table 1: Comparison between result by difference of gain a of conventional MF-NN and result by the novel MF-NN

branch from A to B	Nonlinear Resistance	Gain 0.5	Gain 1.0	Gain 1.5	Gain 2.0	Gain 3.0	PWL function	Conventional MF algorithm
S → 1	60.00 (Ω)	46.0696 (A)	56.5124 (A)	59.2834 (A)	59.8621(A)	59.9952(A)	60.00 (A)	60
S → 3	10.00 (Ω)	9.4537 (A)	9.9741 (A)	9.9990 (A)	10.00(A)	10.00(A)	10.00 (A)	10
1 → 2	30.00 (Ω)	25.7011 (A)	28.8279 (A)	29.7504 (A)	29.9516(A)	29.9983(A)	30.00 (A)	30
1 → 3	20.00 (Ω)	16.7911 (A)	19.1932 (A)	19.8376 (A)	19.9689(A)	19.9989(A)	20.00 (A)	20
1 → 4	10.00 (Ω)	9.3493 (A)	9.8921 (A)	9.9810 (A)	9.9965(A)	9.9999(A)	10.00 (A)	10
2 → 1	20.00 (Ω)	2.8659 (A)	0.7814 (A)	0.1664 (A)	0.03229(A)	0.0011(A)	0.00 (A)	0
2 → 3	20.00 (Ω)	9.3346 (A)	9.8332 (A)	10.1213 (A)	10.1939(A)	10.2113(A)	2.35 (A)	0
2 → 4	10.00 (Ω)	7.0615 (A)	7.8841 (A)	8.1540 (A)	8.2131(A)	8.2269(A)	7.65 (A)	10
3 → 1	10.00 (Ω)	1.6045 (A)	0.4034 (A)	0.0812 (A)	0.0155(A)	0.0005(A)	0.00 (A)	0
3 → 2	10.00 (Ω)	5.3327 (A)	5.0834 (A)	4.9393 (A)	4.9031(A)	4.8943(A)	0.00 (A)	0
3 → 4	50.00 (Ω)	36.6514 (A)	39.6960 (A)	40.5860 (A)	40.7776(A)	40.8218(A)	32.35 (A)	30
4 → 1	20.00 (Ω)	1.3015 (A)	0.2159 (A)	0.0379 (A)	0.0070(A)	0.0002(A)	0.00 (A)	0
4 → 2	20.00 (Ω)	5.8770 (A)	4.2319 (A)	3.6920 (A)	3.5737(A)	3.5463(A)	0.00 (A)	0
4 → 3	30.00 (Ω)	8.0092 (A)	6.1824 (A)	5.6484 (A)	5.5334(A)	5.5069(A)	0.00 (A)	0
2 → T	20.00 (Ω)	17.6488 (A)	19.6446 (A)	19.9399 (A)	19.9889(A)	19.9996(A)	20.00 (A)	20
4 → T	50.00 (Ω)	37.8745 (A)	46.8420 (A)	49.3426 (A)	49.8731(A)	49.9952(A)	50.00 (A)	50
Maximum Flow	$f(s,1) + f(s,3)$	55.5233(A)	66.4866(A)	69.28244(A)	69.8621(A)	69.9952(A)	70.00(A)	70.00

Table 2: Calculation frequency until converging

Gain 0.5	Gain 1.0	Gain 1.5	Gain 2.0	Gain 3.0	PWL function
122(times)	137(times)	244(times)	809(times)	10000(times)	66(times)
567.3(ms)	624.72(ms)	1134.6(ms)	3761.85(ms)	46500(ms)	47.85(ms)

5. Conclusions

In this paper, a novel MF-NN using PWL function was proposed. Simulation results indicated that convergence of the sigmoidal function of conventional MF-NN greatly influences at accuracy and the convergence time of the solution. The error margin with the correct maximum flow became small by raising the gain a. However, there was a fault that takes a lot of computation time. On the other hand, the equivalent result of the maximum flow algorithm was able to be obtained by using novel MF-NN using PWL function. In addition, the computation time was sped up very much. Along with making of more large-scale network, the error margin of the maximum flow and the computation time of conventional MF-NN become larger problems. Therefore, novel MF-NN has been improved to a very superior algorithm.

Acknowlegment

This research was supported by funding from the Open Research Center Project of the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of the Japanese Government(2007-2011)

References

[1] R. K. Ahuja, T. L. Maganti, and J. B. Orlin, "Network Flows:Theory, Algorithms and Applications" Englewood Cliffs, NJ: Prentice-Hall, 1993.

[2] R. Ahlswede, N. Cai, S.-Y. Li, and R. W. Yeung, "Network information flow", IEEE Trans. Inf. Theory, vol. 46, no. 4, pp. 1204-1216, Jul. 2000.

[3] V. Ramachandran, "The Complexity of Minimum Cut and Maximum Flow Problems in an Acyclic Network", Networks, vol. 17, pp. 387-392, 1987.

[4] L. R. Ford, Jr. and D. R. Fulkerson, "Maximal Flow Through a Network", Canadian Journal of Mathematics, pp. 399-404, 1956.

[5] L. R. Ford, Jr. and D. R. Fulkerson, "A Simple Algorithm for Finding Maximal Network Flows and an Application to the Hitchcock Problem", Canadian Journal of Mathematics, pp. 210-218, 1957.

[6] B. V. Cherkassky and A. V. Goldberg, "On implementing the push-relabeled method for the maximum flow problem", Algorithmica, vol. 19, pp. 390-410, 1997.

[7] A. V. Goldberg and R. E. Tarjan, "A new approach to the maximum flow problem", Journal of the ACM, vol. 35, pp. 921-940, 1988.

[8] M. Sato, H. Aomori, and M. Tanaka, "Maximum-Flow Neural Network : A Novel Neural Network for the Maximum Flow Problem", IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol. E92-A, No. 4, pp. 945-951, Apr 2009.

[9] J. J. Hopfield and D. W. Tank, "Neural computation of decisions in optimization problems,", Biolog. Cybern., vol. 52, no. 3, pp. 141-152, 1985.

[10] D. W. Tank and J. J. Hopfield, "Simple neural optimization networks, an A/D converter, signal decision circuit, and a linear programming circuit", IEEE Trans. Circuits Syst., vol. 33, pp. 533-541, May 1986.