"0," "3," "4," "7," "6," "5," "9," "8," "1,"} respectively. The figure shows the circuit operating correctly, initially more than one neuron responds to a particular input, the off-surround feedback will drive down all but the neuron with the greatest response. Checks on the network outputs for a full set of input vectors have shown that the correct neuron response to each input is achieved.

VII. Conclusions

Results showing the effectiveness of the new neural network have been presented and hence the justification for a hardware implementation of this network to enable fast data classification. The DSFPN can train 405 times faster than the commonly used BPN, while still retaining a high recognition performance.

Wavelet descriptor have been shown to give considerably more accurate recognition results as compared to data presented as Fourier descriptors. The error rate reduction was from 2.22% for Fourier descriptors, to only 2.08% for wavelets.

A circuit implementation, for the DSFPN competitive middle layer, has been presented. Simulation results for this circuit show that it can perform reliable pattern recognition at a rate of 100 kHz.

References

[1] J. A. Freeman and D. M. Skapura, *Neural Networks Algorithms Applications and Programming Techniques.* New York: Addison-Wesley, 1992.

[2] I. P. Morns and S. S. Dlay, "Character recognition using Fourier descriptors and a new form of dynamic semi-supervised neural network," *Microelectronics J.* Oxford, U.K.: Elseviere, 1997, vol. 28, no. 1, pp. 73–84.

[3] P. Wunsch and A. F. Laine, "Wavelet descriptors for multiresolution recognition of handprinted characters," *Pattern Recognition.* Oxford, U.K.: Pergamon, 1995, vol. 28, no. 8, pp. 1237–1249.

Solving Graph Algorithms with Networks of Spiking Neurons

Dorel M. Sala and Krzysztof J. Cios

***Abstract*— Spatio-temporal coding that combines spatial constraints with temporal sequencing is of great interest to brain-like circuit modelers. In this paper we present some new ideas of how these types of circuits can self-organize. We introduce a temporal correlation rule based on the time difference between the firings of neurons. With the aid of this rule we show an analogy between a graph and a network of spiking neurons. The shortest path, clustering based on the nearest neighbor, and the minimal spanning tree algorithms are solved using the proposed approach.**

***Index Terms*—Graph algorithms, minimal spanning tree, neural networks, shortest path, spiking neurons.**

I. Introduction

Artificial neural networks (ANN's) are usually represented in the form of a graph. This relationship has been explored by some researchers in the field. A classical example is the traveling salesman problem [2]. More recently, Bose and Liang [1] overviewed both, networks using average firing rate neuron models and graph theory. However, with the rising interest in networks of spiking neurons, which take into consideration timing information, the relationship between graphs and neural networks can be shown from a different perspective. It is our aim to show that. Below we introduce the building blocks of a network of spiking neurons.

Manuscript received September 16, 1998; revised March 20, 1999.

The authors are with the University of Toledo, Toledo, OH 43606 USA.

Publisher Item Identifier S 1045-9227(99)05828-2.

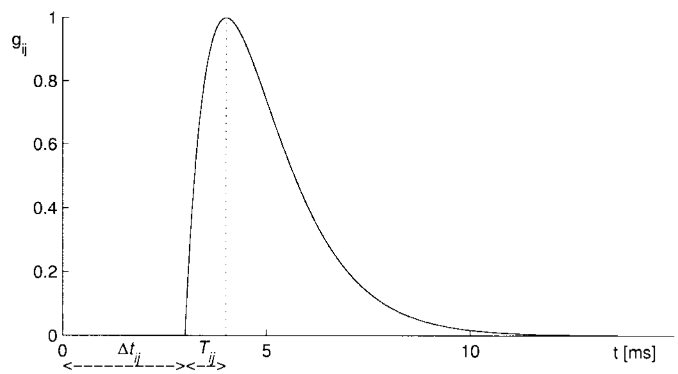

Fig. 1. Normalized postsynaptic potential.

A. Spiking Neuron Model

The spiking neuron model used in our experiments is described by equations defining the properties of the spiking neuron, including equations for transmembrane potential generator, refractory properties, and threshold accommodation [6]

$$\frac{dE}{dt} = \frac{-E + (SC + G_k * (E_k - E) + \sum G_i * (E_i - E))}{T_{\text{mem}}}$$
$$\frac{dT_h}{dt} = \frac{-(T_h - T_{h0}) + c * E}{T_{th}}$$
$$\frac{dG_k}{dt} = \frac{-G_k + B * S}{T_{gk}}$$

where E is the transmembrane potential, T_h is the time-varying threshold, G_k is the potassium conductance, SC is the synaptic current input, E_k is the membrane resting potential, G_i's are the synaptic conductances, E_i's are the synaptic resting potentials, T_{mem} is the membrane time constant, and T_{h0} is the resting value of threshold. Parameter $c \in [0, 1]$ determines the rise of the threshold, T_{th} is the time constant of decay of threshold, parameter B determines the amount of the postfiring potassium increment, T_{gk} is the time constant of decay of G_k, S is one when neuron fires, and is zero otherwise.

B. Synaptic Potentials and Synaptic Plasticity

The postsynaptic potentials (PSP's) are modeled through equivalent synaptic conductance changes [10]. For each synapse between neuron i and j the postsynaptic potential is described by an alpha function [4]

$$g_{ij}(t) = \frac{t - \Delta t_{ij}}{T_{ij}} \cdot \exp\left(\frac{t - \Delta t_{ij}}{T_{ij}}\right) \cdot \exp(1)$$

where Δt_{ij} is the propagation time, T_{ij} is the time constant of the synapse and $\exp(1)$ is a normalizing factor. Fig. 1 shows the shape of a postsynaptic potential. We assume that a synapse is activated whenever a spike generated by the presynaptic neuron reaches it. The exact time of the activation takes place when g_{ij} reaches its peak. Hence the delay between the firing of the presynaptic neuron and

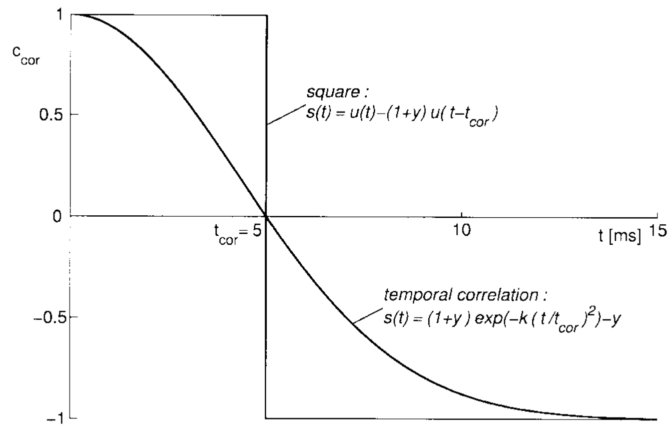

Fig. 2. Different shapes of function $s(t)$; $u(t)$ is the step function, $t_{\rm cor}$ is the correlation time (5 ms), y is the maximum negative excursion (1), and $k = \ln(1 + 1/y)$.

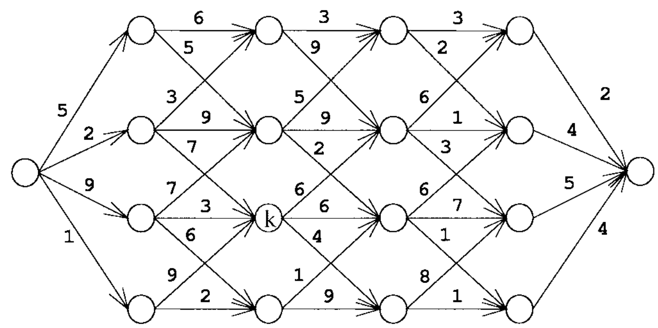

Fig. 3. Directed graph for the shortest path problem. The input is the leftmost node.

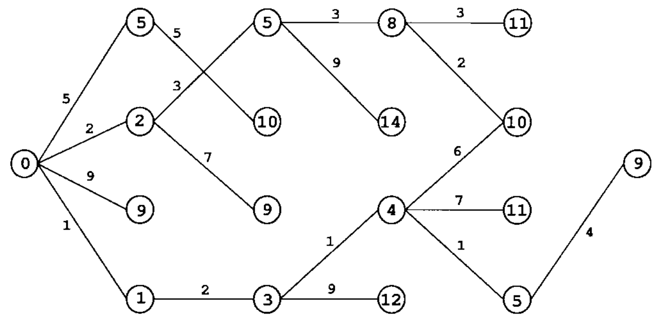

Fig. 4. The resulting network; only the surviving connections are shown; the values inside the circles represent the shortest paths from the input.

the activation of a synapse is $t_{ij} = \Delta t_{ij} + T_{ij}$. In our experiments T_{ij} is the same for all synapses and has a small value such that the change of the firing time of a neuron caused by the modification of its weights can be neglected in calculations.

The total synaptic conductance of a neuron is obtained by summing up all postsynaptic potentials

$$G_j(t) = \sum_i w_{ij} \cdot g_{ij}(t)$$

where w_{ij} represents the strength (weight) of the synapse between neurons i and j.

Learning is done by adjusting the synaptic strength, w_{ij}. This value has an upper bound (M) that represents the saturation value (see Fig. 5). Whenever a synapse is activated its weight is adjusted according to the following rule:

$$w_{ij}(t+1) = w_{ij}(t) \cdot (1 + \alpha \cdot s(t))$$

where α is the learning rate ($\alpha < 1$) and $s(t)$ is a function describing the correlation between neurons i and j. Fig. 2 shows two shapes of the correlation function. The square function determines a fixed increment for the positive part of $s(t)$ and a fixed decrement for the rest. The temporal correlation function [8] provides for a variable increment and decrement, respectively. In the following we use only the temporal correlation function.

II. Shortest Path Through a Graph

Finding the shortest path in a graph is one of the simplest graph algorithms. To show how this can be done with a network of spiking neurons we shall use as an example the directed graph shown in Fig. 3. This graph can be interpreted as a neural network where each circle represents a neuron node and each edge (we consider only positive edges) represents a connection between two neuron nodes. A neuron node consists of a pair of excitatory-inhibitory neurons arranged in such a way that each firing of the excitatory neuron activates the paired inhibitory neuron, which in turn inhibits the excitatory neuron for a period of time that is longer than the longest path in a given graph. With this arrangement each neuron node, henceforth called neuron, will fire only once. The value shown on each edge represents the delay, t_{ij}, between the firing of neuron i and the time the synapse between neurons i and j is activated (in ms).

Initially, the weights of all synapses are equal and sufficiently large so that a single postsynaptic potential can evoke a spike in the postsynaptic neuron. Learning is done in cycles. A cycle starts by initiating a spike in the “input” neuron (in our example the leftmost neuron) and ends when all the neurons in the network have fired. After a number of cycles, that depends on the value of the learning rate, some connections become stronger, up to the saturation value M, while other connections decrease and disappear altogether. Redrawing the network as shown in Fig. 4, we observe that the network represents the initial network in which only the shortest paths from the input neuron to all other neurons have survived. Further examination shows that each neuron has only one entry that has survived, with one exception, namely the neuron at the second row and fifth column in Fig. 4. If we calculate the lengths of the two surviving pathways reaching this neuron, they result in the same value (10). If we were not interested in obtaining multiple solutions we could introduce small perturbations affecting the equal but different paths and eliminate all but one.

Some observations regarding the learning process follow. Whenever a PSP arrives at a synapse that synapse is activated and learning takes place. The first synapse activated is the one that determines the firing of the postsynaptic neuron. If a synapse is activated within a time interval $t_{\rm cor}$, calculated from the time of firing of the postsynaptic neuron, its weight increases, otherwise it decreases. For a small $t_{\rm cor}$ only the first activated synapse will be rewarded and the remaining ones will be penalized. If the learning rate is close to the value of one then the shortest path is obtained in one cycle.

In general one starts with the correlation time, $t_{\rm cor}$, having a large value that is decreased as the learning progresses. By doing that more information regarding the evolution of the weights is available. For example, let us consider the gray-shaded node, k, shown in Fig. 3. It has three input edges of lengths 7, 3, and 9. The evolution of the corresponding synapses in the network is shown in Fig. 5. We notice that the edge labeled 7 is the one corresponding to the shortest path to neuron k, while the path through the edge labeled 9 is the next

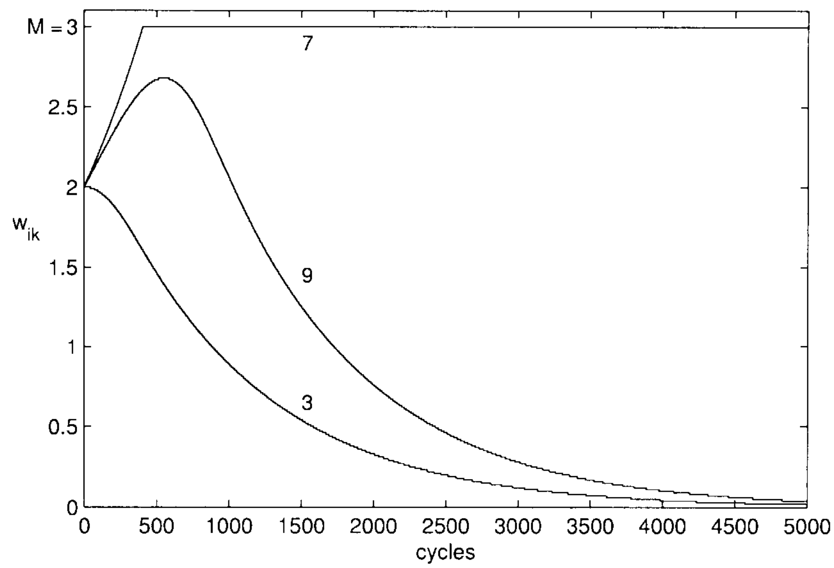

Fig. 5. Input weights evolution of the gray shaded neuron, k, from Fig. 3. The curves are labeled using the edge length of each input; t_{cor} starts at 3 ms and exponentially decays toward zero.

shortest. The path through the edge labeled 3 is the longest. More information can be inferred if we take into consideration the evolution of t_{cor} which starts at 3 ms and decays exponentially toward zero. We notice that path labeled 3 is at least three units longer than the optimal. This type of analysis may be useful for large networks where the shortest path, next to the shortest path, etc., may be difficult to determine otherwise.

III. Clustering

Suppose we are interested in finding how the nodes of a given graph cluster together based on their edge distances. Let us consider the graph shown in Fig. 6, and associate with it a network with two layers of spiking neurons as shown in Fig. 7. The connections between the layers are fixed and allow the activation of a top layer neuron when a bottom-layer neuron is activated. The top-layer neurons are connected in the same way as in the above example, with propagation times given by the edge values, and with their strengths initially set to a large value. The propagation time between the layers is the same; in calculations we used zero. At each step of learning all bottom-layer neurons are stimulated simultaneously with the exception of one neuron, which is chosen randomly. To this neuron corresponds a top-layer neuron, whose firing is triggered by the closest neuron in the top layer. In this way this particular neuron learns its membership to the closest cluster. Learning is cyclical. At the beginning of learning t_{cor} has a large value and decreases with time. When neurons in the top layer fire simultaneously they stimulate each other. If the propagation delay between them is larger than t_{cor} their corresponding synapses decrease, which is an unwanted effect. A sufficient condition to alleviate this effect is to choose the maximum negative excursion in the learning rule (see Fig. 2), $y < 1/n$, where n is the total number of neurons in the layer.

The clustering takes place when the connection strength between two neurons becomes saturated. Fig. 8 shows the evolution of the weights, where white is the strongest. The shortest edges saturate the fastest and the process continues until each neuron is clustered with at least one other neuron. We notice that the network clustering emulates the nearest neighbor algorithm. In our example it resulted in two clusters as pictured in Fig. 9 where only the surviving edges are shown.

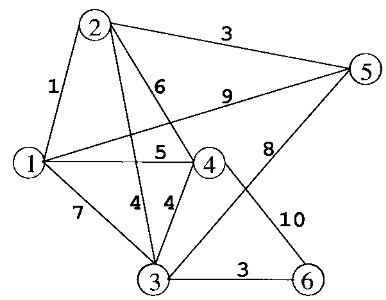

Fig. 6. Bidirectional graph used for clustering.

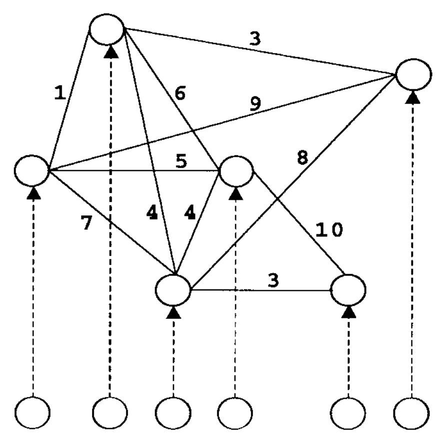

Fig. 7. Two layer network of spiking neurons corresponding to the graph shown in Fig. 6.

It is easy to show that the surviving edges in the clustered network are a subset of the minimal spanning tree. In general, to obtain the minimal spanning tree we need to add one more phase in which the obtained clusters play the role of single neurons. The temporal correlation learning rule will still apply. Now, however, instead of choosing a neuron as input a cluster is selected. This means that all the neurons within that cluster are activated simultaneously at the beginning of a learning cycle. The rest of the process continues in

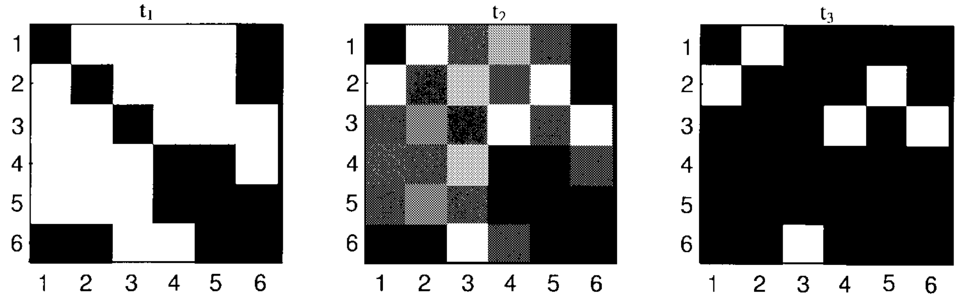

Fig. 8. Evolution of the connection weights in time (the representation is done as an incidence matrix and the neurons are labeled as in Fig. 6; white corresponds to largest value and black to zero). Initially all weights are equal. At t_1 after a few cycles they start to differ, at t_2 the shortest connections are clearly visible, and at t_3 only the shortest connections survive.

TABLE I
KOHENEN'S 32 5-D PATTERNS

		Category																															
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	1	2	3	4	5	6
Attribute	a_1	1	2	3	4	5	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	a_2	0	0	0	0	0	1	2	3	4	5	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	a_3	0	0	0	0	0	0	0	0	0	0	1	2	3	4	5	6	7	8	3	3	3	3	6	6	6	6	6	6	6	6	6	6
	a_4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3	4	1	2	3	4	2	2	2	2	2	2
	a_5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3	4	5	6

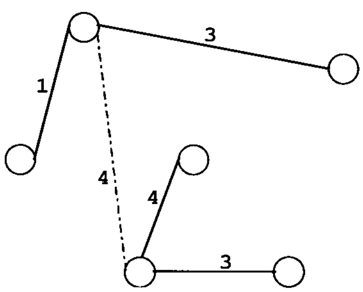

Fig. 9. The clustering process results in two cluster; the shortest edge between the two clusters shown as a dotted line completes the minimal spanning tree.

exactly the same way as for the original clustering phase described above. Thus, as our network has clustered into two clusters of three neurons each, at each step we activate simultaneously the three neurons of a cluster. The neurons from the remaining cluster will be activated, in the order of their connection delays, and the strength of connections will be modified according to the temporal correlation learning rule. Repeating the cycle for a number of times we obtain the minimal spanning tree shown in Fig. 9.

The above observations gave us an idea how to interpret numerical data. To illustrate it we shall use an example consisting of 32 5-D patterns shown in Table I [3]. First, we construct a graph having 32 nodes representing the 32 pattern vectors, with the edge values calculated as the distances between the vectors. We have used the Euclidean distance, although any other distance measure could be used. Next, we associate with this graph a two-layer network of 2 $\times$ 32 spiking neurons. In this context we can say that the distance is coded as the time delay [5], [9]. Applying the same procedure as in the example above clusters start to form, as the learning progresses, as shown by the evolution of the connections strengths in Fig. 10. Initially all connection weights are equal. Starting with a large value for $t_{\rm cor}$ allows the network to strengthen the short-delay connections as shown on the left in Fig. 10. During the learning process $t_{\rm cor}$ decreases to values smaller than most of the delays such that the strengths of all the connections, but the shortest ones, decrease as shown on the right in Fig. 10. Eventually only the shortest-delay connections survive. The white squares in Fig. 10 represent the shortest-delay connections. Drawing the resulting network to show the formed clusters we obtain the network shown in Fig. 11, where the neurons have been labeled according to Table I. In this example just one cluster was obtained, which forms the minimal spanning tree.

All the values of the parameters used in the simulations are given in Table II.

IV. Possible Biological Interpretation

The examples presented above allow for a possible biological interpretation of the results. Results of the first example show that the shortest paths are the ones most probably to survive and dominate. This observation is significant for interpreting results where spiking neurons are used to simulate sensory systems [7]. It suggests that not only the mere power of an input signal is a determinant factor but also its pathway (delays) to upper levels.

The examples used to show the clustering behavior of spiking neurons suggest how association in the higher cortical areas could be achieved. Firing of the neurons can be associated with certain feature(s), object(s), etc. If the features have some common properties they will appear very close together, but not necessarily simultaneously, at some level. Thus, they are associated together based on their relative time cooccurrence. All the examples seem to suggest that the

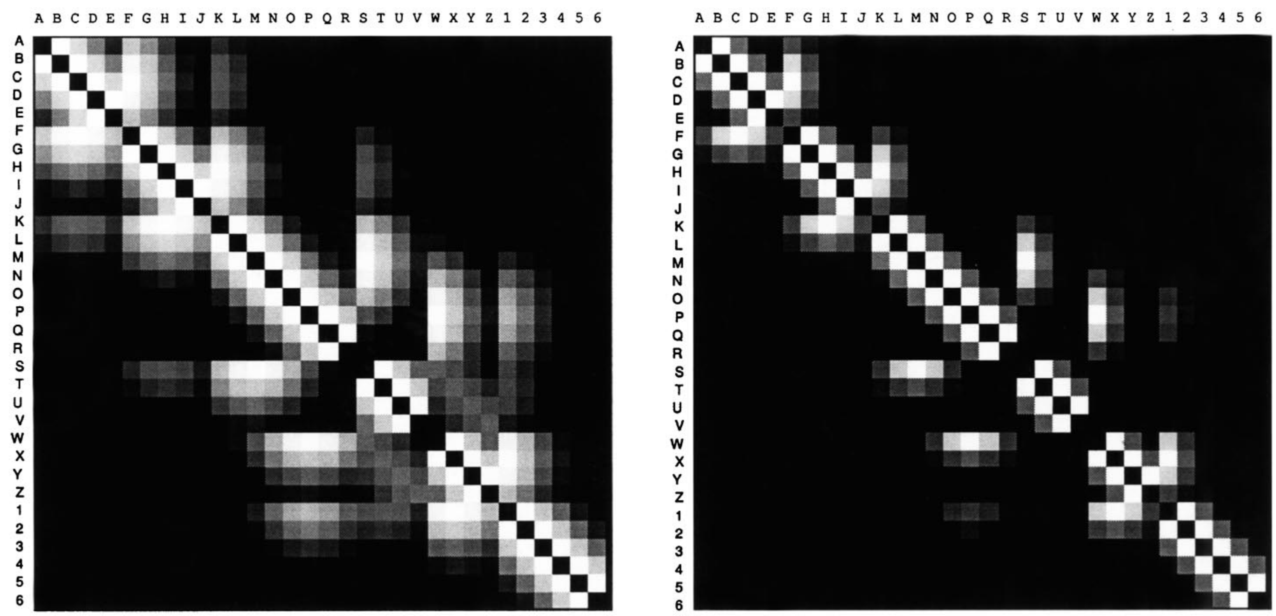

Fig. 10. The connection strengths evolution in time; the weight representations are normalized to the maximum value at that moment of time. To find out the connection strength of a neuron with another neuron we look at the intersection of the row and column labeled as in Table I. The white squares represent maximum strength.

TABLE II
PARAMETER VALUES USED IN SIMULATIONS

E_i		E_k (mV)	c	T_{h0} (ms)	T_{th} (ms)	B	T_{gk} (ms)	T_{mem} (ms)	SC	T_{tj} (ms)
Excitatory (mV)	Inhibitory (mV)									
70	-10	-10	0.2	10	25	20	3	10	0	1

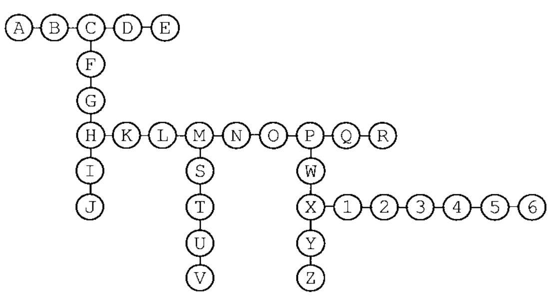

Fig. 11. The resulting network from the Kohonen's example. The neurons are labeled with their corresponding patterns from Table I.

shortest path example is characteristic for the connections between the layers and the clustering example is characteristic for learning within the layers.

V. CONCLUSIONS

We showed how a temporal correlation learning rule implements self-organization in a network of spiking neurons. By considering temporal coding, that represents distances on a given spatial configuration we have shown how to associate graphs with networks of spiking neurons. By using the temporal correlation learning rule we solved the shortest path algorithm, clustering based on the nearest neighbor, and the minimal spanning tree. The examples illustrated that a network of spiking neurons "naturally" emulated those graph algorithms. Finally, possible biological interpretation of this behavior was provided.

REFERENCES

[1] N. K. Bose and P. Liang, *Neural Network Fundamentals with Graphs, Algorithms, and Applications.* New York: McGraw-Hill, 1996.

[2] J. J. Hopfield and D. W. Tank, "Neural computations of decisions in optimization problems," *Biol. Cybern.,* vol. 52, pp. 141–152, 1985.

[3] T. Kohonen, *Self-Organizing Maps.* New York: Springer-Verlag, 1995.

[4] R. Kuhn and J. L. Hemmen, "Temporal association," in *Models of Neural Networks I,* E. Domany, J. L. van Hemmen, and K. Schulten, Eds. New York: Springer-Verlag, 1995, pp. 201–218.

[5] K. T. Judd and K. Aihara, "A neural network model that uses temporal coding by action potentials," *Neural Networks,* vol. 6, pp. 203–215, 1993.

[6] R. J. MacGregor, *Neural and Brain Modeling.* Reading, MA: Academic, 1987.

[7] D. M. Sala, K. J. Cios, and J. T. Wall, "A spatio-temporal computer model of dynamic organization properties of the adult primate somatosensory system," in *Progress in Connectionist-Based Information Systems,* N. Kasabov *et al.,* Eds., vol. I. New York: Springer-Verlag, 1997, pp. 153–156.

[8] D. M. Sala and K. J. Cios, "Self-organization in networks of spiking neurons," *Australian J. Intell., Inform. Processing Syst.,* vol. 5, no. 3, pp. 161–170, 1999.

[9] T. J. Sejnowski, "Time for a new neural code?," *Nature,* vol. 376, pp. 21–22.

[10] J. Xing and G. L. Gerstein, "Networks with lateral connectivity—I: Dynamic properties mediated by the balance of intrinsic excitation and inhibition," *J. Neurophysiol.,* vol. 75, no. 1, pp. 184–199, 1996.