Paper presented at "NATO Advanced Research Workshop on Neurocomputing", Les Arcs, Feb. 1989. 89-C1

An Analog VLSI Architecture for Large Neural Networks

Michel Verleysen, Paul Jespers

Université Catholique de Louvain
Microelectronics Laboratory
3, pl. du Levant
1348 Louvain-la-Neuve, Belgium

Introduction

Neural networks seem to be very interesting in problems such as optimization, pattern recognition and many tasks where perception is more important than a huge amount of computations. However, the interesting properties (speed, fault-tolerance, convergence,...) do not exist when these networks are simulated on conventional computers; the need for dedicated VLSI chips is thus obvious, but the problem is to realize chips where a great number of neurons and synapses can be connected together and integrated on the same chip.

Since a neural network consists in a set of simple processors (neurons) connected to a great number of synapses, it would be difficult to realize them with digital electronics: a simple 100-neuron fully interconnected network would require a hundred 100-input adders, and the area occupied by these cells would be too large for practical applications. Analog techniques will then be used, in order to reduce the size of some elements, even despite a loss of precision.

Analog VLSI architecture for neural networks with synaptic current sources

In artificial neural networks, the elementary processing elements (called neurons by analogy with biological models) are connected through a programmable coupling network. In the model proposed here, each neuron receives information about the state of all the other neurons, weighed by connection strengths (Hopfield's fully interconnected network [1]). A connection between two neurons (called a synapse) can be either excitatory (positive) or inhibitory (negative). Each synapse can source or sink current to the input line of the connected neuron; the direction of the current is determined by a combination of the output of the neuron to which the synapse is connected and a weight value memorized into each synapse. In our model, the output of each neuron is boolean: the logical function of a neuron is thus only a weighed sum of the other neurons with a sign discrimination. Furthermore, only three different connection values are allowed in each synapse: +1, 0 and -1. The direction of the current results from the product of these two values (the neuron output and the connection strength): a product equal to

Publié dans :

NATO ASI Series, Vol. F 68
Neurocomputing
Edited by F. Fogelman Soulié and J. Hérault

1 means a sourced current, to -1 a sunk one and to 0 no connection. The architecture proposed here can be generalized if more than three different states are required.

One simple way to realize this architecture is the solution proposed by De Vegvar and Graf [2]; in this circuit, the synaptic currents are either sourced by a P-type transistor or sunk by an N-type one (Fig. 1). However, this solution presents a major drawback: the matching of currents through the N- and P-type transistors is impaired by the mobility differences between electrons and holes. Since in our model, the logical function of each neuron is to detect the sign of the sum of the sourced and sunk currents, the total mismatching between the P- and N-type transistors on each input line must not exceed one single synaptic current. As the synaptic current itself has to be small enough to allow several single currents to be summed on the same line, mismatching constitutes a strong limitation to the maximum number of neurons that can be put together on the same chip.

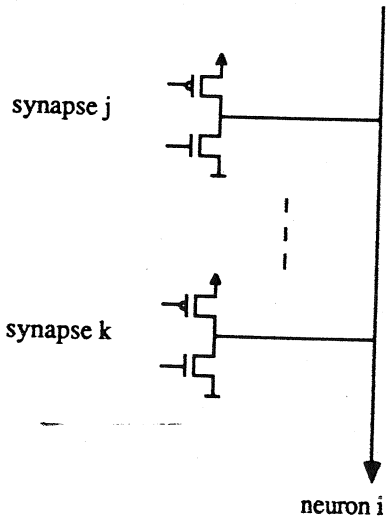

Figure 1. Synapses with P-type and N-type current sources

The problem can be avoided by using the same type of transistors to sink and source current on the input line of the neuron. To realize this, we used two distinct lines to sum the currents: one for excitatory currents, the other for inhibitory currents.

Each synapse is a programmable current source controlling a differential pair (Fig. 2). Three connection values are allowed in each synapse. If "mem1"= 1, current is delivered to one of the two lines with the sign of the connection determined by the product of "mem2" and the output of the neuron to which the synapse is connected. If "mem1"= 0, no connection exists between neurons i and j, and no current flows either to the excitatory line or to the inhibitory line.

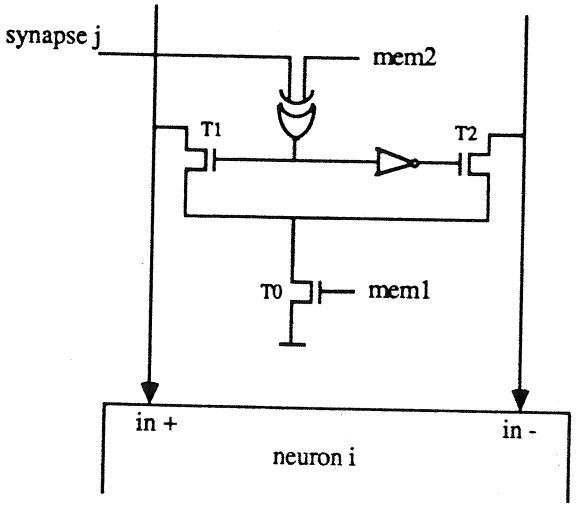

Figure 2. Synapse

Depending on the state of the XOR function, the current may be sourced either on the line i+ or on the line i-. In the neuron, the comparison of the two total currents on the lines i+ and i- must be achieved. This is done by means of the current reflector shown in Fig. 3. The currents on the lines are converted into voltages across transistors *T3* and *T4*; these voltages themselves are compared in the differential input reflector formed by transistors *T5* to *T9*. Because of the two-stage architecture of the neuron, the gain may be very large and the output (*out*) is either 5V if the current in *neuron i-* is greater than the one in *neuron i+*, or OV in the opposite case. The digital output of the neuron can then be fed back into the synapses.

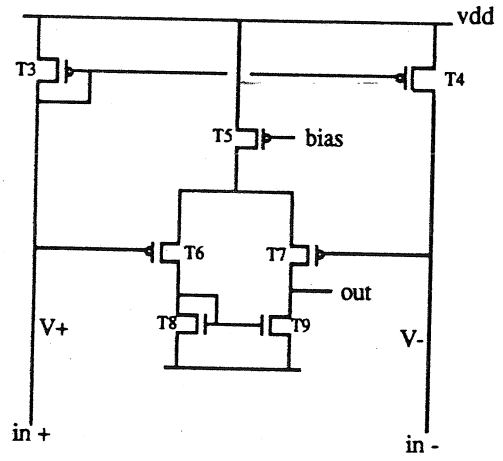

Figure 3. Neuron i

Conclusion

The VLSI architecture for Hopfield's fully interconnected networks proposed here uses very simple neurons and synapses in order to reduce the area occupied by these cells when integrated in a VLSI chip. This will permit the integration of large networks, with several hundreds of fully interconnected neurons. Advantage can be taken from the reduced precision of the synaptic weights by using dedicated learning algorithms [3]. The same architecture can also be used for any type of neural networks: only the connections between neurons and synapses will change, but the architecture of the cells can remain identical.

References

1. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proc. Natl. Acad. Sci. USA, vol.79, pp.2554-2558 (April 1982)
2. Graf, H.P., de Vegvar, P.: A CMOS implementation of a neural network model. Proc. Stanford Conf. on Advanced Research in VLSI, P.Losleben (ed.), MIT Press (1987)
3. Verleysen, M., Sirletti, B., Vandemeulebroecke, A., Jespers, P.: Neural networks for high-storage content-addressable memories: VLSI circuit and learning algorithm. To be published in IEEE Journal of Solid-State Circuits (June 1989)