Learning Capacitive Weights in Analog CMOS Neural Networks

H.C. CARD, C.R. SCHNEIDER AND R.S. SCHNEIDER
Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, Manitoba, Canada R3T2N2

Received July 22, 1991; Revised April 5, 1992

Abstract. Implementations of artificial neural networks as analog VLSI circuits differ in their method of synaptic weight storage (digital weights, analog EEPROMs, or capacitive weights) and in whether learning is performed locally at the synapses or off-chip. In this paper, we explain the principles of analog networks with in situ or local synaptic learning of capacitive weights, with test results of CMOS implementations from our laboratory. Synapses for both simple Hebbian and mean field networks are investigated. Synaptic weights may be refreshed by periodic rehearsal on the training data, which compensates for temperature drift or other nonstationarity. Compact high-performance layouts have been obtained in which learning adjusts for component variability.

Introduction

In order to exploit the efficiencies of artificial neural network (ANN) models it is desirable to build custom analog VLSI circuits which capture both their structure and their learning rules. The mainstream of research in analog VLSI ANNs has employed off-chip learning and reconfigurable topology, and has been successful in developing practical applications for these chips [1, 2]. These chips have the advantage that new developments in improved learning algorithms may be exploited, which is especially important in many current applications of ANNs. At the same time, it is our belief that in special purpose applications, including those typified by nonstationary environments, chips with in situ learning may also play a useful role. In situ learning implies that the learning algorithm is implemented by circuitry local to the individual adaptive synapses. This circuitry computes the weight changes of the synapse during the normal operation of the network. These networks are therefore time variant as well as nonlinear. In this paper we focus on a particular style of analog CMOS network with in situ learning, in which the synaptic weights are represented by charge storage on capacitors.

There are two broad classes of ANN models as shown in Fig. 1. These may be described as feedforward and feedback models. In the feedforward case, during the classification of an input pattern, information flows unidirectionally from input to output through the various network layers [3]. A variation on the feedforward case allows for crosstalk connections within a given layer, as shown by the dashed connection in Fig. 1(a); networks of this type include competitive or Kohonen learning models [4]. In the feedback case, there are also recurrent connections, and the information flows in both directions. In the latter case the network exhibits dynamics, and there is a settling time for the network after the presentation of a new input pattern [5]. These are often called relaxation networks. It is also possible under certain conditions for the system to be unstable, and to enter into a cyclic or oscillatory pattern.

In Fig. 1, the circles are the artificial neurons, which in analog circuits are usually implemented as nonlinear (saturating) amplifiers. The arrows between the neurons represent weighted connections or synapses. A common form of the saturation characteristic of the neuron is a sigmoid given by either the logistic function (for $0 < V_i < 1$)

$$V_i(x_i) = \frac{1}{1 + \exp(-|x_i|)}$$

or the tanh function (for $-1 < V_i < 1$)

$$V_i(x_i) = A \tanh(x_i)$$

These functions are used to model the nonlinear behavior of the synapses in the network.
where the input to the neuron \( x_i \) is the weighted sum of its inputs

\[
x_i = \sum W_{ij} V_j - \Theta_i
\]

with \( W_{ij} \) the weight of the \( ij \) synapse, with \( V_i \) and \( \Theta_i \) the activity and the threshold of neuron \( i \) respectively, and where the summation is over all \( j \) except \( j = i \).

The weights of the synapses are updated according to a learning algorithm. Two common versions of the learning rule are Hebbian learning [6]

\[
\frac{dW_{ij}}{dt} = \epsilon V_j V_i
\]

or alternatively the delta rule [3]

\[
\frac{dW_{ij}}{dt} = \epsilon V_j \delta V_i
\]

where \( \epsilon \) is a learning rate parameter and in the final expression \( \delta V_i \) is an error term. There are often other terms in the learning rules for weight decay, etc. but we do not discuss them here.

The connection weights may be realized in a variety of ways which include: (i) in a hybrid (mixed digital-analog) implementation of a network it is common to store the weights in a binary register, but to perform the synaptic weighting and neural summation with analog circuitry, as for example in [7]; (ii) in a fully analog implementation, the weights may be represented as the charge on a capacitor at each synaptic site, for example [8]; (iii) the weights may be stored as the charge on the floating gate of (analog) EEPROM devices as in [9].

Most VLSI implementations of analog ANNs have run the learning algorithms off-chip in a host processor, and have simply loaded the resulting weights into the synapses in preparation for running the classification task. We are primarily concerned in this paper with more literal implementations, in which circuitry to compute the weight updates associated with learning is incorporated directly into the synapses.

We confine our attention to supervised learning tasks, in which a set of training patterns (input-output examples) is shown to the network. The differences between the outputs generated by the present weights and the desired outputs indicated by the training patterns is then used to derive the required weight changes. Having learned the training set, the network can be shown novel input patterns and may be expected to produce the appropriate outputs for these patterns. This depends on the generalization properties of ANNs, which derives from the ability of the weights to interpolate in the input space between the examples from the training set. The present work is complementary to the sensory preprocessing of Mead, et al. [10] in that our emphasis is on synaptic learning; also our circuits do not operate principally in the transistor subthreshold regime.

**Synaptic Weights Represented as Charge on Capacitors**

We now describe analog ANNs with weights represented by the voltage or charge on capacitors associated with the individual synapses, as in Fig. 2. Weight updating of capacitors employs relatively simple analog circuitry, leading to com-