Available online at www.sciencedirect.com
SCIENCE DIRECT
Neural Networks 18 (2005) 179–189
Neural Networks
www.elsevier.com/locate/neunet

NEUROM: a ROM based RNS digital neuron

Giuseppe Alia[a,*], Enrico Martinelli[b]

[a]Dipartimento di Ingegneria dell'Informazione, Università di Pisa, Via Caruso, 56100 Pisa, Italy
[b]Dipartimento di Ingegneria dell'Informazione, Università di Siena, Via Roma, 56, 53100 Siena, Italy

Received 8 July 2002; accepted 11 November 2004

Abstract

In this work, a fast digital device is defined, which is customized to implement an artificial neuron. Its high computational speed is obtained by mapping data from floating point to integer residue representation, and by computing neuron functions through residue arithmetic operations, with the use of table look-up techniques. Specifically, the logic design of a residue neuron is described and complexity figures of area occupancy and time consumption of the proposed device are derived. The approach was applied to the logic design of a residue neuron with 12 inputs and with a Residue Number System defined in such a way as to attain an accuracy better than or equal to the accuracy of a 20-bit floating point system. The proposed design (NEUROM) exploits the RNS carry independence property to speed up computations, in addition it is very suitable for using look-up tables. The response time of our device is about $8 \times T_{ACC}$, where T_{ACC} is the ROM access time. With a value of T_{ACC} close to the 10 ns allowed by the current ROM technology, the proposed neuron responds within 80 ns, NEUROM is therefore the neuron device proposed in the literature which allows for maximum throughput. Moreover, when a pipeline mode of operation is adopted, the pipeline delay can assume a value as low as about 14 ns. In the case study considered, the total amount of ROM is about 5.55 Mbits. Thus, using current technology, it is possible to integrate several residue neurons into a single VLSI chip, thereby enhancing chip throughput. The paper also discusses how this amount of memory could be reduced, at the expense of the response time.

Keywords: Artificial neuron; Residue Number Systems; Look-up table; VLSI; Pipeline operation; Floating point notation; Fast processing; ROM

1. Introduction

For many years, interest in Artificial Neural Networks (ANN) has been growing in a broad area of applications. A great deal of effort has been made to enhance ANN performance, especially in the framework of very complex applications with real time constraints, for example, real time image processing and artificial vision (Bello, 2000; Clark & Furth, 1999; Hammadou, Bouzerdoum, Bermak, & Eshraghian, 2000; Montufar-Chaveznava, Guinea, Garcia-Alegre, & Preciado, 2001). As far as computing support for an advanced ANN emulation is concerned, the approaches proposed in the literature range from general purpose superscalar computers down to special parallel systems (Ming-Jung, Hau, & Vijayan, 2003), as well as to very fast special processors. There are several examples of the last approach in the literature. In (Clarkson, Ng, & Guan, 1993) a special VLSI chip is presented which implements a p-RAM. In (Bolouri, Morgan, & Gurney, 1994) a neural system, HyperNet, is proposed, which is based on five VLSI custom IC's. In (El-Mousa & Clarkson, 1996), a neuro-computer board is presented, which incorporates the pRAM-256 VLSI neural processor. Another interesting example is a board containing four SIMD-type processors named NEUR04, which can efficiently support a general neural network (Komori, Arima, Kondo, Tsubota, Tanaka, & Kyuma, 1997). In (Lau, 1996) high performance chips/systems for processing neural networks are presented, among which there are two fully digital implementations, namely the floating point SNAP (SIMD Numerical Array Processor) and the fixed point CNAPS (Connectionist Neural Adaptive Processor System). In (Hasan & Ng, 1997) the SLiFBAM, a parallel VLSI BAM processor, is described, which is based on bidirectional associative

* Corresponding author. Tel.: +39 50 2217 511; fax: +39 50 2217 522.
E-mail addresses: g.alia@iet.unipi.it (G. Alia), enrico@dii.unisi.it (E. Martinelli).

0893-6080/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2004.11.006

memories. There are many research papers related to hardware implementation of neural networks. However, in order to make significant comparisons, only those proposals that strictly relate to the goals of our work are considered in this paper.

We focus on the definition of a fast digital device that implements an artificial neuron. More specifically, with reference to a trained neural network, a neuron is considered in which inputs and weights are real numbers. Our approach is to enhance its computational speed implementing it by means of a physical device based on residue arithmetic units. The proposed approach exploits both the RNS carry independence property to speed up the computations, and the ability of the RNS to be implemented by means of lookup tables.

In Section 2 the fundamentals of a Residue Number System (RNS), which is the number representation system adopted in this work, are recalled.

In Section 3 the integer reduction problem in an artificial neuron is considered, and the aim is to find the integer range required to emulate neural computations that use floating point notation, without loss of accuracy.

In Section 4 the logic design of a table lookup structure for a residue neuron is described, while Section 5 outlines a case study concerning the design of a residue neuron covering the requirements of many neural applications in terms of both input number and variables ranges.

With regard to design issues, a 12-input neuron is considered, as well as a 20-bit floating point notation, with 16 bits of signed mantissa and 4 bits of two-complement exponent, is assumed for all variables modelling the neuron. With these values, several applications can achieve the required level of accuracy and a ROM-based implementation (NEUROM) yields a high time performance and a practicable area occupancy. In fact, we evaluated the response time of this residue neuron to be less than 80 ns, while a total memory amount of about 5.55 Mbits is required. It will be shown that NEUROM has a better throughput than other high performance devices proposed in the literature. Moreover, the feasibility of integrating several NEUROMs on the same chip will be discussed.

The number of NEUROM inputs can be increased, provided that the overall number of ROMs is increased as well, without ROM address space increasing. On the other hand, the upper bound to the address space expansion limits the size of RNS moduli.

Finally, when a pipeline mode of operation is adopted, we have evaluated that the pipeline delay can assume a value close to the time necessary to access a ROM.

Memory-based implementations in the field of Pattern Recognition were first proposed in (Bledsoe & Blisson, 1962; Bledsoe & Browning, 1959), and memory-based neurons theory was assessed in (Gurney, 1989) and (Gorse & Taylor, 1988, 1990a,b,c, 1991; Gorse, Taylor, & Clarkson, 1993, 1997).

1.1. Nomenclature

In the following a nomenclature of less known mathematical terms and symbols is given. However, symbols will generally also be defined in the text when they first occur.

$\lfloor x \rfloor$	is the greatest integer less than or equal to the real number x
$\lceil x \rceil$	is the least integer greater than or equal to the real number x
abs(x)	is the absolute value of x
$\lvert n \rvert_m$	with n and m integers, denotes the remainder of n/m that is $n - m\lfloor n/m \rfloor$
$[A,B]$	is the set of numbers between A and B, with $A < B$, A and B included
$[A,B)$	is the set of numbers between A and B, with $A < B$, A included, B not included
$\{e_i\colon \mathrm{P}(e_i)\}$	is the set of elements e_i that satisfy predicate $\mathrm{P}(e_i)$
$\{e_i\}$, $i=1,\ldots,n$	is a set of n elements e_i
$\{e_1, e_2, e_3\}$	is the set of elements e_1, e_2, e_3
$\leftrightarrow$	states a one-to-one correspondence between an integer and its RNS representation
sign(x)	sign of x
a^*	is the real number corresponding to the floating point representation of the real number a
$\approx$	approximate equality

2. Residue Number Systems

In a Residue Number System, a set of p integers $\{m_j\colon m_j > 1\}$ called *moduli* is defined as the representation base. In such a system, an integer N is represented by means of p integers n_j, where $n_j = \lvert N \rvert_{m_j} = N - \lfloor N/m_j \rfloor m_j$; in this equality, $\lfloor N/m_j \rfloor$ denotes the largest integer not exceeding N/m_j. Integers n_j are called *residue digits*. If moduli m_j are pairwise relatively prime, it can be shown (Szabo & Tanaka, 1967) that there is a unique representation for each number N in the range $[0,M)$, with $M = \prod_{j=1}^{p} m_j$.

Moreover, for any pair of integers N_1 and N_2, the relations

$$\lvert N_1 \pm N_2 \rvert_{m_j} = \lvert \lvert N_1 \rvert_{m_j} \pm \lvert N_2 \rvert_{m_j} \rvert_{m_j},$$

$$\lvert N_1 N_2 \rvert_{m_j} = \lvert \lvert N_1 \rvert_{m_j} \lvert N_2 \rvert_{m_j} \rvert_{m_j}$$

hold. Namely, any residue digit of a sum and of a product can be obtained from the corresponding residue digits of operands and, consequently, addition and multiplication can be performed separately for each modulus.

It is easy to show that previous representation technique can be extended to signed arithmetic. In fact, an implicit