Available online at www.sciencedirect.com
SCIENCE DIRECT
Neural Networks 18 (2005) 896–913
Neural Networks
www.elsevier.com/locate/neunet

Generalized hamming networks and applications

Konstantinos Koutroumbas[a,*], Nicholas Kalouptsidis[b]

[a]Institute for Space Applications and Remote Sensing, National Observatory of Athens, Metaxa and V. Pavlou, Palaia Penteli, 15236, Athens, Greece
[b]Department of Informatics and Telecommunications, Division of Communications and Signal Processing, University of Athens, Athens, Greece

Received 15 December 2003; accepted 21 February 2005

Abstract

In this paper the classical Hamming network is generalized in various ways. First, for the Hamming maxnet, a generalized model is proposed, which covers under its umbrella most of the existing versions of the Hamming Maxnet. The network dynamics are time varying while the commonly used ramp function may be replaced by a much more general non-linear function. Also, the weight parameters of the network are time varying. A detailed convergence analysis is provided. A bound on the number of iterations required for convergence is derived and its distribution functions are given for the cases where the initial values of the nodes of the Hamming maxnet stem from the uniform and the peak distributions. Stabilization mechanisms aiming to prevent the node(s) with the maximum initial value diverging to infinity or decaying to zero are described. Simulations demonstrate the advantages of the proposed extension. Also, a rough comparison between the proposed generalized scheme as well as the original Hamming maxnet and its variants is carried out in terms of the time required for convergence, in hardware implementations. Finally, the other two parts of the Hamming network, namely the competitors generating module and the decoding module, are briefly considered in the framework of various applications such as classification/clustering, vector quantization and function optimization.
© 2005 Elsevier Ltd. All rights reserved.

Keywords: Non-linear hamming networks; Mathematical and computational analysis; Hamming Maxnet; GEMNET; HITNET; SELECTRON; Competitors generating module; Decoding module; Clustering; Classification; Quantization; Function optimization

1. Introduction

The Hamming network, was originally presented in (Steinbuch, 1961; Steinbuch & Piske, 1963; Taylor, 1964) as an associative memory model and has come into the scene again through (Baum, Moody, & Wilczek, 1987; Lippmann, 1987). Since then, it has been studied by several researchers. The Hamming network as an associative memory network has been analyzed in Floreen (1991).

The architecture of the Hamming network is shown in Fig. 1. It consists of three modules. The competitors generating module (CGM) is content dependent and produces a set S of M non-negative numbers that compete with each other. For example, in the associative memory and vector quantization context, these numbers are the matching scores between an input vector to the network and each one of the M stored vectors. In the function optimization context, these numbers are the values of the function at given points. The decoding module (DM) takes as input the output of the Hamming maxnet and shapes it so as to match the requirements of the application at hand. The CGM and DM modules are briefly discussed in the framework of certain application areas at the end of the present work.

In the present paper we first focus on the second module of the Hamming network, the Hamming maxnet (HMN) (see Fig. 2). The HMN consists of two layers of nodes. The first layer includes M nodes and has a recurrent structure (i.e. each node takes input from all the nodes of the same layer including itself). The initial state is formed by the outputs of the competitors generating module. Once convergence is established, it reads out an M-dimensional vector having all coordinates equal to zero except one which is positive and corresponds to the node with the maximum initial value.

* Corresponding author. Tel.: +30 210 8109 189.
E-mail addresses: koutroum@space.noa.gr (K. Koutroumbas), kalou@di.uoa.gr (N. Kalouptsidis).

0893-6080/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2005.02.003

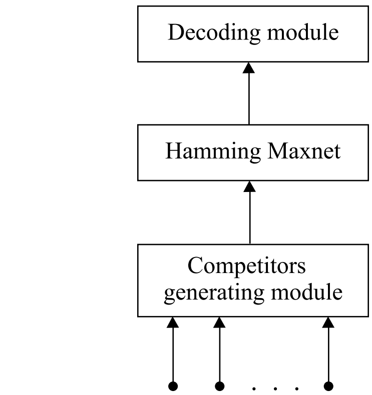

Fig. 1. The general architecture of the Hamming maxnet.

The *state of a node at time t* is denoted by $x_i(t)$, $i=0,\ldots,M-1$. In its original version, the weight from a node to itself equals 1 and the weight from a node to any other node equals $-\varepsilon$, where ε is chosen to be a positive constant less than $1/(M-1)$ (Koutroumbas & Kalouptsidis, 1994; Lippmann, 1987). The nodes are updated via the following equation

$$x_i(t+1) = f(H_i(t+1)), \quad i=0,\ldots,M-1 \tag{1}$$

where

$$H_i(t+1) = x_i(t) - \varepsilon \sum_{m=0,\, m\neq i}^{M-1} x_m(t), \quad i=0,\ldots,M-1, \tag{2}$$

and f is the ramp function, i.e. $f(x)=\max(x,0)$. A prerequisite for reliable operation of the HMN is to have at least one node with positive initial value and only a single node with maximum initial value. In addition, Eq. (2) indicates that the *parallel mode of operation* is employed, that is all nodes are updated simultaneously. Different versions of the HMN where not all nodes are updated synchronously are discussed in (Koutroumbas, 1995; Koutroumbas and Kalouptsidis, 1994). In the extreme case where each node is updated asynchronously to the others, it is guaranteed that the network identifies only one of possibly more nodes with maximum initial value.

The second layer of the HMN consists of M threshold nodes each fed by the output of the corresponding node of the previous layer. A node of this layer outputs 1 if its input is positive and 0 otherwise.

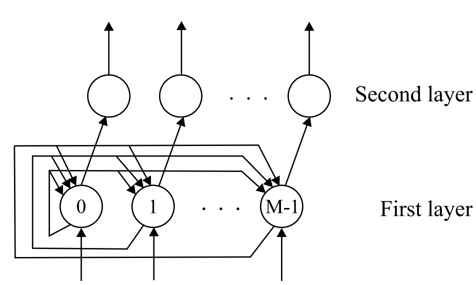

Fig. 2. The original Hamming maxnet.

In Koutroumbas and Kalouptsidis (1994) a detailed theoretical analysis of both parallel and partially parallel modes of operation of the HMN is provided. Various generalizations of the multiplying coefficients are discussed in Koutroumbas (1995). Stabilization methods in conjuction with accelerating convergence techniques are also discussed in Koutroumbas (1995). The complete solution of the HMN is provided for the fully parallel mode of operation in Sum and Tam (1996).

Hardware implementations of the Hamming networks are presented in He, Cilingiroglu, and Sinencio (1993) and Robinson, Yoneda, and Sinencio (1992). Finally, efficient implementations of recurrent networks on silicon are developed in Hahnloser, Sarpeshkar, Mahoward, Douglas, and Seung (2000).

One of the major drawbacks of the HMN is its slow convergence. Significant effort has been undertaken by several researchers to address this issue. In this spirit a modified version of the HMN that is much faster than the original one is discussed in Yen and Chang (1992) and Yadid-Pecht and Gur (1995). In this case, ε does not remain constant but is adjusted at each iteration as follows: $\varepsilon(t)=1/M(t)$, where $M(t)$ is the number of the nodes with positive value at the tth iteration.

Alternative methods for accelerating convergence of HMN in terms of the number of iterations at the cost of increased computational complexity per iteration are developed in Yang, Chen, Wang, and Lee (1995) and Yang and Chen (2000). The resulting networks are referred to as GEMNET and HITNET. The main idea here is to subtract from each node the highest possible value from all nodes. Such values are determined on the basis of assumptions on the probability distribution that generates the members of S. These methods achieve further improvement compared to the above discussed modified version of HMN, in terms of the number of iterations. However, as far as the HITNET is concerned, there is always the possibility of over-inhibition,[1] even in cases where the original maxnet would succeed. The dynamics of GEMNET and HITNET are given by

$$x_i(t+1) = f\left(\gamma x_i(t) - \frac{\gamma}{M(t)/\beta(t) - 1} \sum_{j=0}^{M-1} x_j(t),\right) \quad i=0,\ldots,M-1. \tag{3}$$

f and $M(t)$ are defined as above, $\gamma \geq 1$ and $\beta(t)$ is greater than 1 for HITNET and equal to 1 for GEMNET.

A weakness of the original HMN as well as its variants discussed above, is their inability to work properly when there are two or more nodes with maximum initial value. This situation is met frequently in applications where the elements of S stem from a discrete domain. For example, in the associative memory application M binary vectors, $\boldsymbol{y}_i$,

[1] The case where all nodes are led to zero after a finite number of iterations.

Download English Version:

https://daneshyari.com/en/article/10325987

Download Persian Version:

https://daneshyari.com/article/10325987

Daneshyari.com