Influence of Self-Connection Weights on Cellular-Neural Network Stability

Sergey Pudov

Supercomputer Software Department
Computing Center of Siberian Branch
Russian Academy of Science
Pr. Lavrentieva, 6, Novosibirsk, 630090, Russia
E-mail: pudov@ssd.sscc.ru

Abstract. Cellular-Neural Associative Memory (memory by Hopfield with local connection structure) with weight matrix designed by anyone of the existing methods ensuring individual stability of network is considered. It is studied how self-connection weight values influence the main characteristic of CNAM, namely the strong stability to k-distortions of stored prototypes. Expression for determining the self-connection weight values is obtained, such that provides a maximal strong stability for each prototype. Two strategies are proposed to determine the most acceptable value according to the required accuracy. The obtained results are valid not only for CNAM but also for full-connected Hopfield associative memory designed with the help of any learning method.

1 Introduction

Cellular-Neural Associative Memory (CNAM) is a Neural Associative Memory (NAM) by Hopfield [1] with connection structure like that of Cellular Automaton. The completeness of NAM's connection graph is a grave obstacle to VLSI-implementation and makes extremely complex the retrieval algorithm. Hence, the restriction of connection number is very attractive from this point of view, but it accordingly decreases the storing and computing capability of neural network. Moreover, most of existing learning methods developed for Hopfield network are not applicable for CNAM, because they are based on solving matrix equations, which result in a full-complete weight matrix [2,3]. If the amount of connections is to be restricted, the unwanted connections are simply cut [4]. This approach decreases both storing capability and stability of the network. The latter is due to the fact, that the relation between self-connection weight and that of the rest of connections is changed. It is clear, that CNAM learning method should take into account connection structure of the network. Such a method has been suggested by the author in [5], where CNAM storing capacity is achieved to be about 2 * q patterns (q – a number of neuron connection), the stability being increased as compared with Rosenblatt's perceptron learning rule.

In [5] it was also shown by simulation that the addition of self-connections increases 7 – 10 times the capability to recognize distorted patterns for a CNAM, which stored symbols drawn in thin lines. A similar influence of self-connection
weight was shown in [6] for full-connection networks learned by Projection Learning Rule [2]. From this it follows that self-connections have a great significance and it makes sense to try to improve some characteristics of neural network by self-connection weights correction, particularly to increase its stability.

In this paper self-connection influence on the capability of a CNAM to restore distorted patterns is investigated. It is also shown how to calculate self-connection weights which result in the largest stability increase of an already learned CNAM. Formulas for precise and approximate calculations are given. This paper is organized as follows. In Section 2 formal representation of CNAM is given. In Section 3 and 4 the self-connection weight for the cases of 1-distortions and of k-distortions respectively is obtained.

2 Formal representation of Cellular-Neural Associative Memory

Following [7] CNAM is defined by three notions: \( N = \langle C, W, \Phi \rangle \), where \( C \) is a rectangular array consisting of \( m \) rows and \( n \) columns of cells (or neurons) with states \( c_{ij} \in \{-1, 1\} \). \( W = \{W_{ij}\} \) is a set of weight vectors in a form \( W_{ij} = (w_1, \ldots, w_q) \), \( w_k \) being a real number assigned to the connection between a neuron with coordinates \((i, j)\) and its \( k \)-th neighbor. \( \Phi \) is a rule according to which CNAM acts.

The set of coordinate pairs forms naming space \( M \) on which naming functions \( \phi(i, j) : M \to M \) are defined. The set of all cellular arrays with the same naming space \( M \) is referred to as \( C(M) \). For each cell \((i, j)\) a set of other cells which communicate with it forms its neighborhood being determined by a connection template

\[
T(i, j) = \{\phi_1(i, j), \ldots, \phi_q(i, j)\},
\]

where \( \phi_k(i, j) \neq \phi_l(i, j) \) for any \((i, j) \in M\) and for all \( k, l = 1 \ldots q, k \neq l \).

For an array \( C \in C(M) \) all neighbor states of a neuron \((i,j)\) are represented as a vector \( C_{ij} = (c_{\phi_1(i,j)}, \ldots, c_{\phi_q(i,j)}) \), or for short \( C_{ij} = (c_1, \ldots, c_q) \). Further a vector

\[
D_{ij} = c_{ij} \ast C_{ij} = (d_1, \ldots, d_q),
\]

which is called a normalized neighborhood of neuron \((i,j)\) will be also used. Both vectors \( C_{ij} \) and \( W_{ij} \) have \( q \) components each, the numeration being given birth by that of naming functions in (1), their scalar product being defined as \( \langle C_{ij}, W_{ij} \rangle = \sum_i c_i w_i \).

The rule \( \Phi \) of CNAM operating is described by a following iterative procedure:

**Procedure.** Let \( C(t) \in C(M) \) be the array after the \( t \)-th iteration, \( C_{ij} \) and \( W_{ij} \) - neighbor states and weight vectors of a neuron \((i, j)\) respectively. Then

1. all cells in \( C(t) \) compute the following function: