Available online at www.sciencedirect.com
SCIENCE DIRECT

Future Generation Computer Systems 22 (2006) 441–446

FGCS
FUTURE GENERATION COMPUTER SYSTEMS

www.elsevier.com/locate/fgcs

A differential approach to solve the inverse eigenvalue problem derived from a neural network

N. Del Buono

Dipartimento di Matematica, Università degli Studi di Bari, via E. Orabona 4, I-70125 Bari, Italy

Available online 5 January 2005

Abstract

We consider the problem of designing a general additive neural network which possesses prescribed equilibria. The relation between this design problem and a problem of generating a matrix with specified eiegenvalues, which maps a given set of vectors of another given set, is investigated. The obtained inverse eigenvalue problem is then solved using a gradient flow approach. Working with discretisation of systems of differential equations allows to preserve the original dimension of the problem and could give the possibility of constructing adaptive schemes faster than algebraic one.

Keywords: Inverse eigenvalue problem; Gradient flow approach; Neural network

1. Introduction

Designing an artificial neural network is a key issue in different applications areas, such as for instance, pattern classification, function approximation, optimization and control problems [2]. There are two fundamental tasks for network designers: to locate the equilibria and analyze their stability for an existing network and to construct a network which possesses a prescribed point as stable equilibrium (see [4,5]). Here, this latter question is tackled using a differential approach. In particular, the relation between the design problem an inverse eigenvalue problem in linear algebra is used to construct a cost function and to derive a gradient flow whose limiting solutions solve the specific design problem. The equivalence between the design problem and the inverse eigenvalue one has been shown in [5]. In that paper, some results on the existence of solutions have been proved and a stable algorithm based on pole assignment techniques in control theory has been proposed to compute the solution.

Here, we seek to replace algebraic methods of computing by geometric ones, i.e. via the analysis of a certain system of differential equations whose solutions solve our problem. Working with ordinary differential systems derived from the minimization problem of a well design cost function allows us to preserve the original dimension of the problem (which would be increased if an optimization tool would be applied directly to the minimization problem) and furthermore, permits the use of new algorithms and the possibility

E-mail address: delbuono@dm.uniba.it.

doi:10.1016/j.future.2004.11.023

of constructing adaptive schemes faster then existing algebraic approaches [1,3].

This paper is organized as follows. In the next section, the design problem of an additive neural network which possesses prescribed equilibria is presented. Starting from the set of ordinary differential equations describing the network, we show how to reformulate this design problem as a matrix inverse eigenvalue problem. In Section 3, we recall some known concepts and results on a more general inverse eigenvalue problem and then we propose the cost function to be minimized. In Section 4, we derive the gradient flow system and we prove that its solutions solve the specific task. Finally, in the last section we report some numerical test showing the behavior of our approach.

2. The design problem and the matrix inverse eigenvalue problem

An additive neural network can be modeled by the following set of ordinary differential equations [4]:

$$\dot{u} = -Au + Wg(u) + p, \tag{1}$$

where $u = [u_1, u_2, \ldots, u_n]^{\mathrm{T}}$ is the state vector whose components are the n neurons of the neural network, A is a diagonal real matrix, $g(u) = [g_1(u_1), \ldots, g_n(u_n)]^{\mathrm{T}}$, where the $g_i : \mathbb{R} \to \mathbb{R}$, $i = 1, \ldots, n$, are the nonlinear squashing functions which are strictly increasing and approaching fixed limits for large negative and positive values of u_j, $W = w_{ij}$ is an $n \times n$ real matrix whose elements are the weights connecting the ith and the jth neurons and $p = [p_1, \ldots, p_n]^{\mathrm{T}}$ is the external input vector.

A particular task for neural network designer is to project a network that possesses a given equilibrium, that is how to find the weight matrix W such as the system (1) has a prescribed stable equilibrium $\bar{u}$.

This design problem (DP) can be formulated mathematically as follows: given a diagonal matrix $A \in \mathbb{R}^{n\times n}$, an activation function $g : \mathbb{R}^n \to \mathbb{R}^n$, and two vectors $p, \bar{u} \in \mathbb{R}^n$, find the configuration of the network, i.e. the weight matrix W, such that the network has $\bar{u}$ as an asymptotic stable equilibrium point.

Imposing that (1) has $\bar{u}$ as equilibrium point, then the weight matrix W must satisfy the following equality constraint

$$-A\bar{u} + Wg(\bar{u}) + p = 0. \tag{2}$$

Moreover, since we require that $\bar{u}$ is an asymptotically stable solution, then the eigenvalues of the Jacobian matrix of the system at $\bar{u}$:

$$J = -A + WG, \tag{3}$$

where $G = \mathrm{diag}(g_1'(\bar{u}), \ldots, g_n'(\bar{u}))$, should belong to the left half complex plane.

Taking in mind these conditions, the design problem can be transformed into an equivalent inverse eigenvalue problem (IEP) in the following way.

Being A and G diagonal matrices, from (3) we can easily rewrite the weight matrix W as

$$W = (J + A)G^{-1}. \tag{4}$$

Then, substituting (4) in (2), we get

$$-A\bar{u} + (J + A)G^{-1}g(\bar{u}) + p = 0. \tag{5}$$

Now, let $\mathcal{L} = \{\lambda_1, \ldots, \lambda_n\}$ be an arbitrary set of complex numbers closed under complex conjugation (i.e. $\lambda \in \mathcal{L} \Leftrightarrow \bar{\lambda} \in \mathcal{L}$), and x, y denote the vectors $x = G^{-1}g(\bar{u})$ and $y = A\bar{u} - AG^{-1}g(\bar{u}) - p$, hence the design problem of a neural network can be reformulated as an inverse eigenvalue problem (IEP) in the following way: given two vectors $x, y \in \mathbb{R}^n$ and chosen the set $\mathcal{L}$ so that each element has negative real part, find a matrix J such that $Jx = y$ and $\sigma(J) = \mathcal{L}$, where $\sigma(J)$ denotes the spectrum of the matrix J.

Hence, once solved the IEP associated to the DP, the required weight matrix W can be calculated through (4).

3. A general matrix eigenvalue problem

The inverse eigenvalue problem associated to the design question is the prototype of the following more general matrix inverse eigenvalue problem (MIEP): given two matrices $X, Y \in \mathbb{R}^{n\times p}$, with $p \leq n$ and an arbitrary set $\mathcal{L}$ of complex number closed under complex conjugation, find a $n \times n$ matrix J such that $JX = Y$ and $\sigma(J) = \mathcal{L}$.

The first question arising when one considers an inverse eigenvalue problem is about its solvability. The following result, proved in [6], answers to this question for the general MIEP stated above.

Theorem 1. *Suppose that rank* $([X]) = p$. *Then the MIEP is solvable for any* $\mathcal{L}$ *if the matrix* $Y - sX$ *is a full*