Information Processing Letters 113 (2013) 876–881

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Improved neural solution for the Lyapunov matrix equation based on gradient search

Yuhuan Chen [a], Chenfu Yi [b,c,*], Dengyu Qiao [b]

[a] Center for Educational Technology, Gannan Normal University, Ganzhou 341000, China
[b] Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
[c] School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou 341000, China

ARTICLE INFO

Article history:
Received 25 January 2013
Received in revised form 19 August 2013
Accepted 1 September 2013
Available online 9 September 2013
Communicated by X. Wu

Keywords:
Analysis of algorithms
Recurrent neural networks
Gradient search
Hierarchical identification principle
Energy function
Activation function

ABSTRACT

By using the hierarchical identification principle, based on the conventional gradient search, two neural subsystems are developed and investigated for the online solution of the well-known Lyapunov matrix equation. Theoretical analysis shows that, by using any monotonically-increasing odd activation function, the gradient-based neural networks (GNN) can solve the Lyapunov equation exactly and efficiently. Computer simulation results confirm that the solution of the presented GNN models could globally converge to the solution of the Lyapunov matrix equation. Moreover, when using the power-sigmoid activation functions, the GNN models have superior convergence when compared to linear models.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

The solution of matrix equation is frequently encountered in many scientific and engineering fields. For example, matrix square root is used for the design and application for the Kalman filter [1,2]. Quadratic programming is exploited for the solution of robot manipulators [3,4] and communication [5]. Generally speaking, there are two types of solutions to such problems. One type of solution is the numerical algorithms performed on digital computers. Usually, such numerical algorithms are of serial-processing nature and may not be efficient enough for large-scale online or real-time applications [6]. The second type of solution to this problem is based upon parallel approaches, where many methods have been developed and implemented on specific architectures. The neural-dynamic approach is now viewed as an effective parallel-processing method and a powerful alternative for online computation and optimization.

In recent years, due to the in-depth research in recurrent neural networks (RNN), a variety of computational methods based on neural solvers have been proposed to solve such numerical problems [7–12]. For example, a novel linear matrix inequality-based stability criterion is obtained by using Lyapunov functional theory to guarantee the asymptotic stability of uncertain fuzzy recurrent neural networks with Markovian jumping parameters in [7]. Wang neural networks was proposed to solve the linear simultaneous equation $Ax = b$ [8]. A type of functional neural networks is proposed in [9,10] for the efficient calculation of eigenpairs of a matrix. Zhang neural networks were proposed to solve the Sylvester matrix equation with time-varying coefficient matrices in real-time [12]. Moreover, because of the parallel distributed nature of neural networks and their hardware realizability [11,12,15], recurrent neural networks have been applied widely in many scientific and engineering fields.

* Corresponding author at: Research Center for Biomedical and Information Technology, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China. Tel.: +86 755 86392209.
E-mail address: ltchenve@gmail.com (C. Yi).

0020-0190/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ipl.2013.09.002

In this paper, based on the hierarchical identification principle [16–18], a gradient-based neural network (GNN) is developed, exploited, and investigated for the Lyapunov (or Lyapunov-like) matrix equations widely encountered in many scientific and engineering fields, e.g., control theory [7], linear algebra [16], and boundary value problems [19]. Theoretical proof shows that the improved neural solution can converge to the solution of such Lyapunov equation. The presented illustrative example substantiates the superior convergent performance of the given GNN models as well.

The remainder of this paper is organized as follows. In Section 2, GNN models are developed and analyzed for the solution of $AX = B$, together with its convergence analysis. By using the hierarchical identification principle, Section 3 presents two sub-neural models for the online solution of the constant Lyapunov matrix equation $A^T X(t) + X(t)A = -C$. In Section 4, an example is presented and illustrated for the demonstration the GNN models for such problem solving. Some final remarks about this paper are given in the last section.

Before ending this introductory section, the main contributions of the letter are listed as follows.

(i) In this paper, a type of GNN model is developed and investigated for solving the well-known Lyapunov matrix equation. Theoretical results are also given to show the effectiveness of such neural models.
(ii) By using the hierarchical identification principle, the well-known Lyapunov equation is decomposed into two sub-equations solved by the GNN models, respectively.
(iii) An illustrative example is simulated, compared and discussed to substantiate that the presented GNN models could solve the Lyapunov matrix equation with accuracy and effectiveness.

2. Gradient-based neural networks for linear matrix equation $AX = B$

In this section, a gradient-based recurrent neural networks (GNN) is presented for the online solution of the following linear matrix equation by using the least-square method [20].

$$AX = B, \tag{1}$$

where $A \in R^{n\times n}$ is a nonsingular constant matrix, matrix $B \in R^{n\times n}$ is also time-invariant, and $X(t)$ is an unknown matrix to be solved. Then, such suggested neural models would provide a solution for the Lyapunov matrix equation to be solved in the ensuing section.

2.1. Construction of general GNN model

Generally speaking, a GNN model is constructed by defining a scale-valued norm-based energy function. Evolving along the descent direction resulting from such energy function, the GNN model could obtain the neural solution of the problems; i.e., the minimum point is equal to the solution of Eq. (1). Then, according to this classical gradient search [8,11–14,16,21], a nonnegative scalar-valued norm-based energy function $\varepsilon(x)$ could be firstly defined as follows.

$$\varepsilon(X) := \|AX - B\|_F^2 = \mathrm{trace}\big((AX - B)^T (AX - B)\big), \tag{2}$$

where Frobenius norm $\|A\|_F := \sqrt{\mathrm{trace}(A^T A)}$ [11,12,20,22], of which the minimal point (it is also a global minima here) equals the solution of the problem. In addition, by the following basic differential properties of matrix trace [11,22]:

$$\frac{\partial\, \mathrm{trace}(BAC)}{\partial A} = B^T C^T, \qquad \frac{\partial\, \mathrm{trace}(BA^T C)}{\partial A} = CB, \tag{3}$$

we have the differential equation with respect to X:

$$\frac{\partial \varepsilon(X)}{\partial X} = A^T (AX - B).$$

Then, along with the descent direction of the negative gratitude of such energy function, i.e.,

$$-\frac{\partial \varepsilon(X)}{\partial X} = -A^T (AX - B).$$

Finally, the general linear GNN model for the linear matrix equation (1) could be achieved as follows:

$$\dot{X}(t) = -\Gamma \frac{\partial \varepsilon(X)}{\partial X} = -\gamma\big(A^T (AX - B)\big), \tag{4}$$

where $X(t)$, starting from an initial condition $X_0 := X(0) \in R^{n\times n}$, is the activated neural solution corresponding to the solution $X^*(t) \in R^{n\times n}$ of Eq. (1), the matrix-valued design parameter (or say, learning rate) Γ could simply be γI with constant scalar $\gamma > 0$ and I being an identity matrix. As an inductance parameter or the reciprocal of a capacitance parameter, $\gamma > 0$ should be set as large as the hardware permits and is generally used to scale the convergence rate [23].

As inspired by Zhang et al.'s design method [12,15], we could obtain the solution performance by using different nonlinear activation arrays $\mathcal{F}(\cdot)$ with each element listed in the following form:

(i) linear activation function $f(u) = u$;
(ii) bipolar sigmoid activation function $f(u) = (1 - \exp(-\xi u))/(1 + \exp(-\xi u))$ with $\xi \geqslant 2$;
(iii) power activation function $f(u) = u^p$ with odd integer $p \geqslant 3$ (note that linear activation function $f(u) = u$ can be viewed as a special case of power activation function with power-index $p = 1$); and
(iv) power-sigmoid activation function

$$f(u) = \begin{cases} u^p, & \text{if } |u| \geqslant 1, \\ \frac{1+\exp(-\xi)}{1-\exp(-\xi)} \cdot \frac{1-\exp(-\xi u)}{1+\exp(-\xi u)}, & \text{otherwise}, \end{cases}$$

with suitable design parameters $\xi \geqslant 2$ and $p \geqslant 3$.

Therefore, by using the above-mentioned activation function arrays, the linear GNN model (4) could be transformed to the following general nonlinear dynamical equation:

$$\dot{X}(t) = -\Gamma \frac{\partial \varepsilon(X)}{\partial X} = -\gamma A^T \mathcal{F}(AX - B) \tag{5}$$

Download English Version:

https://daneshyari.com/en/article/427134

Download Persian Version:

https://daneshyari.com/article/427134

Daneshyari.com