Applied Mathematics and Computation 215 (2009) 1301–1306

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Global exponential convergence and stability of gradient-based neural network for online matrix inversion

Yunong Zhang [a,*], Yanyan Shi [a], Ke Chen [a], Chaoli Wang [b]

[a] Department of Electronics and Communication Engineering, Sun Yat-Sen University, Guangzhou 510275, China
[b] Electrical Engineering Faculty, University of Shanghai for Science and Technology, Shanghai 200031, China

ARTICLE INFO

Keywords:
Gradient-based neural network
Online matrix inversion
Lyapunov stability theory
Asymptotical convergence
Global exponential convergence

ABSTRACT

Wang proposed a gradient-based neural network (GNN) to solve online matrix-inverses. Global asymptotical convergence was shown for such a neural network when applied to inverting nonsingular matrices. As compared to the previously-presented asymptotical convergence, this paper investigates more desirable properties of the gradient-based neural network; e.g., global exponential convergence for nonsingular matrix inversion, and global stability even for the singular-matrix case. Illustrative simulation results further demonstrate the theoretical analysis of gradient-based neural network for online matrix inversion.

1. Introduction

The problem of matrix inversion is one of the basic problems widely encountered in science and engineering. It is usually considered to be an essential part of many solutions; e.g., as preliminary steps for optimization [1,2], signal-processing [3–5], electromagnetic systems [6], and robot inverse kinematics [7,8]. Since the mid-1980s, efforts have been directed towards computational aspects of fast matrix inversion and many algorithms have thus been proposed [9–14]. Usually, the minimal arithmetic operations for numerical algorithms are proportional to the cube of the matrix dimension [14], and consequently such algorithms performed on digital computers may not be efficient enough for large-scale online applications. In view of this, some $O(n^2)$-operation algorithms were proposed to remedy this computational problem, e.g., in [4,5]. However, they might still not be fast enough; e.g., in [5], it takes on average around one hour to solve a 60,000-dimensional matrix-inverse problem. As a result, for this online matrix-inversion problem, many parallel-processing computational methods have been developed, analyzed, and implemented on specific architectures [3,7–10,12,15–21].

The dynamic system approach is one of such important parallel-processing methods for solving matrix-inversion problems [3,8,15–20]. Recently, due to the in-depth research in neural networks, numerous dynamic and analog solvers based on recurrent neural networks have been developed and investigated [3,8,16–20]. The neural-dynamic approach is thus regarded as a powerful alternative to online computation because of its parallel distributed nature and convenience of hardware implementation [7,8,15,21,22].

To solve online the matrix-inversion problem, almost all neural networks are designed based on the following defining equation,

$$AX - I = 0, \tag{1}$$

* Corresponding author.
E-mail addresses: ynzhang@ieee.org (Y. Zhang), bluesky_54@163.com (Y. Shi), is03chk@mail2.sysu.edu.cn (K. Chen), clwang@usst.edu.cn (C. Wang).

0096-3003/$ - see front matter
doi:10.1016/j.amc.2009.06.048

where matrix $A \in R^{n\times n}$ should be nonsingular, $I \in R^{n\times n}$ denotes the identity matrix, and $X \in R^{n\times n}$ is the unknown matrix to be obtained. Clearly, when the defining Eq. (1) is solved, X equals the inverse A^{-1}. To solve for A^{-1}, Wang proposed the following recurrent neural network [8,17]:

$$\dot{X}(t) = -\gamma A^T (AX(t) - I), \quad X(0) \in R^{n\times n}, \; t \in [0, +\infty), \tag{2}$$

where design parameter $\gamma > 0$, being an inductance parameter or the reciprocal of a capacitance parameter, is set as large as the hardware permits, or selected appropriately for simulative and/or experimental purposes. When applied to inverting nonsingular matrices, only global asymptotical convergence was proven in [17] for such a gradient-based neural network (2). That is,

Lemma. *Consider a nonsingular matrix $A \in R^{n\times n}$. Equilibrium state $X^* := A^{-1}$ of gradient-based neural network (2) is asymptotically stable in the large.*

However, we know that global exponential convergence (GES) is a much more desirable property of neural-dynamic systems. It implies that a recurrent neural network could converge arbitrarily fast to the theoretical inverse $X^* := A^{-1}$. So, in this paper, we investigate the GES property of gradient-based neural network (2) when applied to nonsingular matrix inversion. For a better understanding on the difference between asymptotical convergence and exponential convergence, we show Fig. 1 which shows the significance of this work as well. In contrast to exponential convergence, a pure asymptotical convergence implies that GNN solution $X(t)$ approaches the theoretical inverse X^* only as time t goes to $+\infty$, which may not be accepted in practice.

The remainder of this paper is thus organized into three sections. Section 2 presents (i) the global exponential convergence of GNN (2) when applied to nonsingular matrix inversion, and (ii) the global stability of GNN (2) when applied to singular matrices. Section 3 gives illustrative simulation-results, which substantiate the aforementioned theoretical analysis. Finally, Section 4 concludes the paper with final remarks.

2. Theoretical analysis

Compared to the previously-proved asymptotical convergence in [8,17], we now provide the following theorem and proof about the global exponential convergence of gradient-based neural network (2) when applied to nonsingular matrix inversion.

Theorem 1. *Consider nonsingular matrix $A \in R^{n\times n}$. Starting from any initial state $X(0) = X_0 \in R^{n\times n}$, state matrix $X(t) \in R^{n\times n}$ of gradient-based neural network (2) will exponentially converge to theoretical inverse $X^* = A^{-1}$. In addition, the exponential convergence rate is the product of γ and the minimum eigenvalue α of A^TA.*

Proof. Let $\widetilde{X}(t) := X(t) - X^*$ denote the difference between the GNN solution $X(t)$ and the theoretical solution X^* of Eq. (1). Thus, we have $X(t) = \widetilde{X}(t) + X^*$ and $\dot{X}(t) = \dot{\widetilde{X}}(t)$. Substituting the above two equations into GNN (2) yields

$$\dot{\widetilde{X}}(t) = -\gamma A^T A \widetilde{X}(t). \tag{3}$$

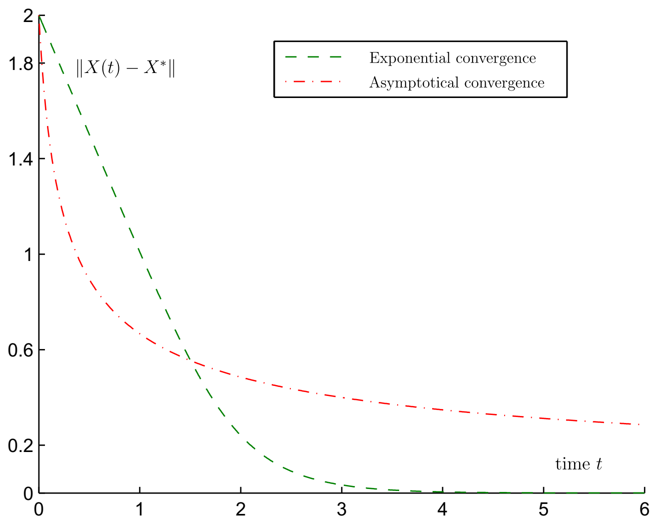


Fig. 1. Comparison between asymptotical convergence and exponential convergence.

Now, we can define a Lyapunov function candidate $E(t)$ as follows:

$$E(t) = \|A\widetilde{X}(t)\|_F^2/2 = \mathrm{tr}([A\widetilde{X}(t)]^T[A\widetilde{X}(t)])/2 = \mathrm{tr}(\widetilde{X}^TA^TA\widetilde{X})/2 \geqslant 0, \quad (4)$$

where $\|A\|_F := \sqrt{\mathrm{tr}(A^TA)}$ denotes the Frobenius norm of matrix A, and $\mathrm{tr}(A)$ generates the trace of matrix A (i.e., the sum of the main diagonal elements of A). Evidently, $E(t)$ is positive definite in the sense that $E(t) > 0$ for any $\widetilde{X}(t) \neq 0$, and $E(t) = 0$ only for $\widetilde{X}(t) = 0$ (which corresponds to equilibrium state $X(t) = X^*$). In addition, when $\|\widetilde{X}\| \to +\infty, E(t) \to +\infty$ as well.

Moreover, we have the time derivative of $E(t)$ along the system trajectory of gradient-based neural network (2) as follows:

$$\begin{aligned}\dot{E}(t) = \frac{dE(t)}{dt} &= \mathrm{tr}\left(\left(\frac{\partial E}{\partial(A\widetilde{X})}\right)^T \frac{d(A\widetilde{X})}{dt}\right) = \mathrm{tr}((A\widetilde{X})^TA\dot{\widetilde{X}}) = \mathrm{tr}((A\widetilde{X})^TA(-\gamma A^TA\widetilde{X})) = -\gamma\mathrm{tr}((A\widetilde{X})^TAA^T(A\widetilde{X})) \\ &= -\gamma\|A^TA\widetilde{X}\|_F^2 \leqslant 0. \end{aligned} \quad (5)$$

Evidently, $\dot{E}(t)$ is negative definite in the sense that $\dot{E}(t) < 0$ for any $\widetilde{X}(t) \neq 0$, and $\dot{E}(t) = 0$ only for $\widetilde{X}(t) = 0$ (corresponding to equilibrium state $X(t) = X^*$), due to A being nonsingular, design parameter $\gamma > 0$ and Eq. (5).

Thus, according to Lyapunov stability theory [8,18,20,23,24], it follows from Eqs. (4) and (5) that equilibrium state $\widetilde{X} = 0$ of (3) is globally asymptotically stable. Since $\widetilde{X}(t) := X(t) - X^*$, we have the following equivalent result: equilibrium point $X^* = A^{-1}$ of GNN (2) is globally asymptotically stable. This actually completes the proof of the lemma in Section 1 by an alternative way instead of linear systems theory used in [17].

More importantly, given $\alpha > 0$ as the minimum eigenvalue of A^TA, it follows from Eq. (5) that

$$\dot{E}(t) = -\gamma\mathrm{tr}\{(A\widetilde{X})^TAA^T(A\widetilde{X})\} \leqslant -\gamma\alpha\mathrm{tr}\{(A\widetilde{X})^T(A\widetilde{X})\} = -\gamma\alpha\|A\widetilde{X}(t)\|_F^2 = -2\gamma\alpha E(t).$$

As a result, $E(t) \leqslant \exp(-2\gamma\alpha t)E(0)$. In addition, we have

$$\begin{aligned} E(t) &= \mathrm{tr}(\widetilde{X}^TA^TA\widetilde{X})/2 \geqslant \alpha\mathrm{tr}(\widetilde{X}^T\widetilde{X})/2 = \alpha\|\widetilde{X}(t)\|_F^2/2, \\ E(0) &= \|A\widetilde{X}(0)\|_F^2 \leqslant \|A\|_F^2\|\widetilde{X}(0)\|_F^2. \end{aligned}$$

The following inequality could thus be derived from the above three equations:

$$\alpha\|\widetilde{X}(t)\|_F^2/2 \leqslant E(t) \leqslant \exp(-2\gamma\alpha t)E(0) \leqslant \exp(-2\gamma\alpha t)\|A\|_F^2\|\widetilde{X}(0)\|_F^2,$$

which could be simplified as

$$\|\widetilde{X}(t)\|_F = \|X(t) - X^*\|_F \leqslant \frac{\|A\|_F\|\widetilde{X}(0)\|_F}{\sqrt{\alpha/2}}\exp(-\alpha\gamma t) := \beta\exp(-\alpha\gamma t),$$

with $\beta := \|A\|_F\|\widetilde{X}(0)\|_F/\sqrt{\alpha/2}$. The proof about the global exponential convergence of gradient-based neural network (2) is thus complete (with convergence rate being $\alpha\gamma$). $\square$

In addition to the above theorem and proof about global exponential convergence for inverting nonsingular matrices, in the remainder of this section, we provide the following theorem and proof about the global stability of gradient-based neural network (2) when applied to singular matrices.

Theorem 2. *Given singular matrix $A \in R^{n\times n}$, gradient-based neural network (2) is still globally stable.*

Proof. We can construct a Lyapunov function candidate as below:

$$E(t) = \|AX - I\|_F^2/2 = \mathrm{tr}((AX - I)^T(AX - I))/2 \geqslant 0. \quad (6)$$

Then, along the system trajectory of gradient-based neural network (2), the time derivative of Lyapunov function candidate $E(t)$ could be obtained

$$\begin{aligned}\dot{E}(t) &= \mathrm{tr}\left(\left(\frac{\partial E}{\partial X}\right)^T\frac{dX}{dt}\right) = \mathrm{tr}((AX - I)^TA\dot{X}) = \mathrm{tr}((AX - I)^TA(-\gamma A^T(AX - I))) = -\gamma\mathrm{tr}((AX - I)^TAA^T(AX - I)) \\ &= -\gamma\mathrm{tr}((A^T(AX - I))^T(A^T(AX - I))) = -\gamma\|A^T(AX - I)\|_F^2 \leqslant 0. \end{aligned} \quad (7)$$

Thus, by Lyapunov stability theory [8,18,20,23,24], gradient-based neural network (2) is globally stable, in view of $\dot{E}(t) < 0$ for any non-equilibrium state X with $A^T(AX(t) - I) \neq 0$, and $\dot{E}(t) = 0$ only for $\dot{X} = 0$ [i.e., at equilibrium state X with $A^T(AX(t) - I) = 0$]. The proof is thus complete. $\square$

3. Simulative examples

In order to demonstrate and substantiate the aforementioned global exponential convergence and global stability of gradient-based neural network (2), in this section, we present computer-simulation results. The simulations are based on MATLAB routine "ode45" [25]. For illustration and comparison, we consider the following nonsingular matrix $A^{(1)}$ and singular matrix $A^{(2)}$:

$$A^{(1)} = \begin{bmatrix} 1 & -2 & 3 \\ 0 & -3 & -2 \\ 1 & 2 & 3 \end{bmatrix}, \quad A^{(2)} = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}.$$

3.1. Nonsingular matrix inversion

Using gradient-based neural network (2) with different values of γ to invert nonsingular matrix $A^{(1)}$, we could have Figs. 2 and 3.

- The former figure shows that starting from randomly-generated initial states, the computational errors $\|AX(t) - I\|_F$ of gradient-based neural network (2) all converge to zero. In addition, the convergence of such errors could be expedited by increasing γ. The larger the γ is, the faster the gradient-based neural network (2) converges to theoretical inverse $X^* = A^{-1}$.
- The latter figure depicts typical convergence of state matrix $X(t)$ of gradient-based neural network (2) to theoretical inverse $X^* = A^{-1}$. Note that, for the above nonsingular matrix $A^{(1)}$, the minimum eigenvalue of A^TA is $\alpha = 0.1716$. Hence, with design parameter $\gamma = 100$, the convergence rate of GNN (2) is $\alpha\gamma = 17.16$, and the convergence time to achieve a tiny solution error of $\exp(-7)\beta = 0.0009\beta$ is approximately 407.9 ms. When design parameter γ is increased to 1000, the convergence rate of GNN (2) is now $\alpha\gamma = 171.6$, and the convergence time to achieve a the same solution-error of 0.0009β is only 40.79 ms. This comparison is shown evidently in Fig. 3.

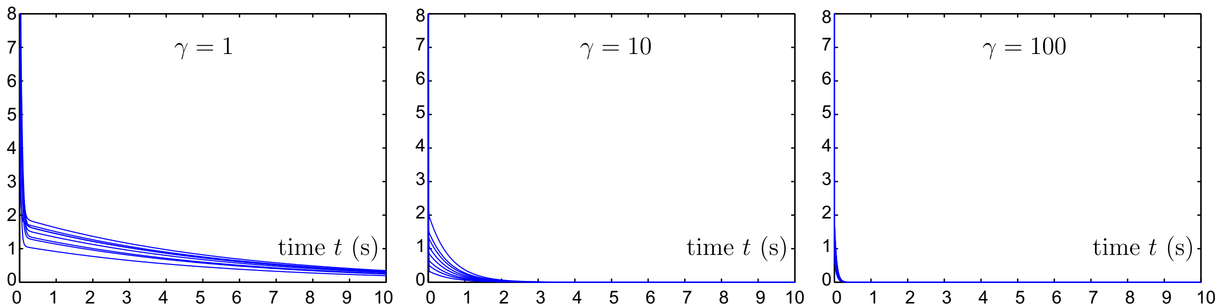


Fig. 2. Global exponential convergence of computational error $\|AX(t) - I\|_F$ synthesized by gradient-based neural network (2) for inverting a nonsingular matrix.

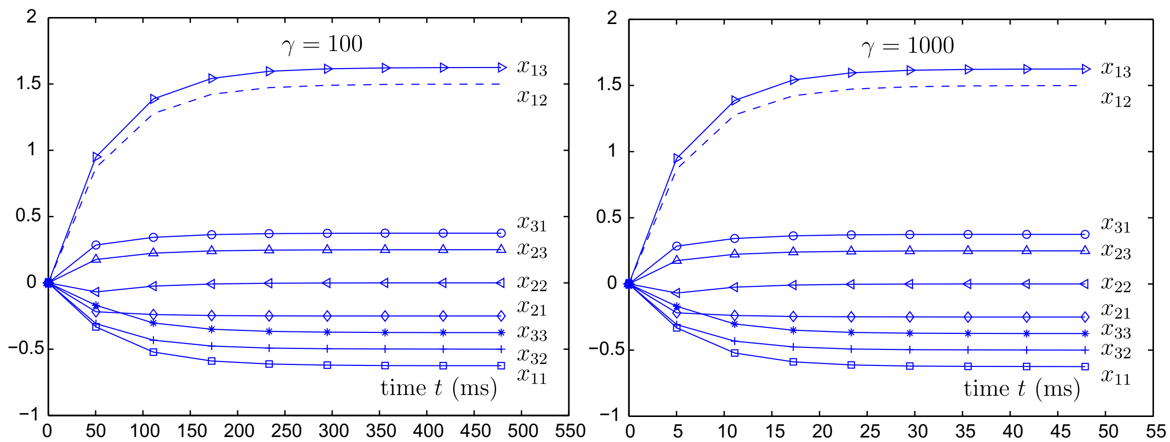
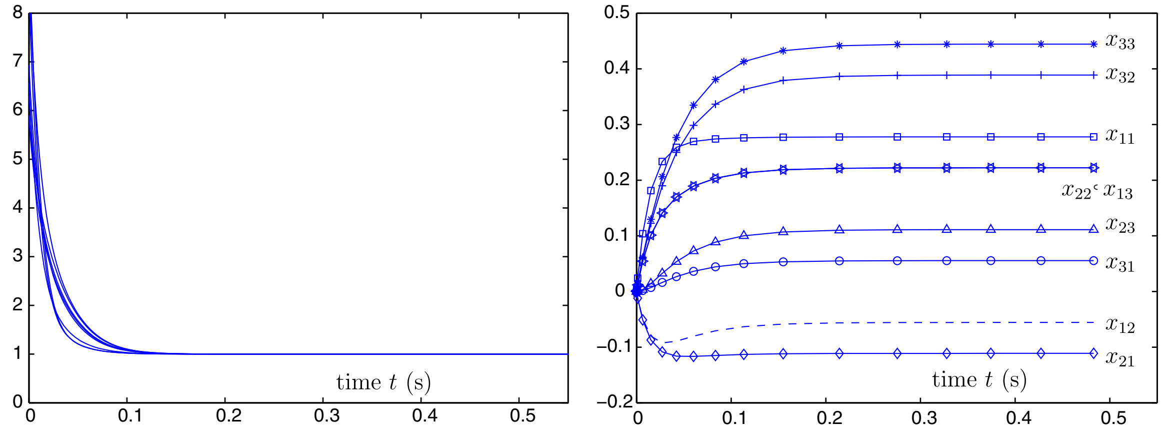


Fig. 3. Exponential convergence of state $X(t)$ of gradient-based neural network (2) starting from zero initial state and with different γ for inverting a nonsingular matrix.



(a) error $\|AX(t) \quad I\|_F$ (b) state $X(t)$

Fig. 4. Global convergence of $\|AX(t) - I\|_F$ illustrates global stability of gradient-based neural network (2) with $\gamma = 10$ when applied to singular matrix $A^{(2)}$.

In summary, the above two figures and observations have substantiated the theoretically-proved global exponential convergence of gradient-based neural network (2) when inverting nonsingular matrices.

3.2. Singular-matrix situation

For singular matrix $A^{(2)}$ given before, it has no inverse and Eq. (1) has no solution. However, even in this singular situation, gradient-based neural network (2) still works well in terms of stability. This is shown in Fig. 4. Specifically, starting from randomly-generated initial states, the Lyapunov function $\|AX(t) - I\|_F$ used in (6) and related to GNN (2) decreases to zero, as time t goes on. In addition, the right graph of Fig. 4 shows a typical situation of state matrix $X(t)$ of gradient-based neural network (2) starting from the zero initial state $X(0) = 0 \in R^{n\times n}$ and converging to an equilibrium state X^*. It is worth mentioning that X^* here equals the MATLAB "pinv" solution [25], which is substantiated by a large number of simulations [8]. Thus, as inspired from this study, for better consistency with prevailing numerical algorithms, we can always use the zero initial state $X(0) = 0 \in R^{n\times n}$ to start the neural network (2).

In summary, the above figure and observations have substantiated the theoretically proved global stability of gradient-based neural network (2) when applied to singular matrices.

4. Conclusions

Wang proposed a gradient-based neural network [i.e., (2)], which could compute online matrix-inverses in a parallel-processing manner. Based on Lyapunov stability theory, this paper has investigated the global exponential convergence and global stability of such a gradient-based neural network for inverting matrices, no matter whether they are invertible or not. Illustrative computer-simulation results have substantiated the theoretical analysis.

Acknowledgements

This work is funded by National Science Foundation of China under Grant 60643004 and by the Science and Technology Office of Sun Yat-Sen University. Before joining Sun Yat-Sen University in 2006, the corresponding author, Yunong Zhang, had been with National University of Ireland, University of Strathclyde, National University of Singapore, Chinese University of Hong Kong, since 1999. He has continued the line of this research, supported by various research fellowships/assistantship. His web-page is now available at http://www.ee.sysu.edu.cn/teacher/detail.asp?sn=129.

References

[1] Y. Zhang, W.E. Leithead, Exploiting Hessian matrix and trust-region algorithm in hyperparameters estimation of Gaussian process, Applied Mathematics and Computation 171 (2005) 1264–1281.
[2] Y. Zhang, Towards piecewise-linear primal neural networks for optimization and redundant robotics, in: Proceedings of IEEE International Conference on Networking, Sensing and Control, 2006, pp. 374–379.
[3] R.J. Steriti, M.A. Fiddy, Regularized image reconstruction using SVD and a neural network method for matrix inversion, IEEE Transactions on Signal Processing 41 (1993) 3074–3077.
[4] W.E. Leithead, Y. Zhang, $O(N^2)$-operation approximation of covariance matrix inverse in Gaussian process regression based on quasi-Newton BFGS method, Communications in Statistics – Simulation and Computation 36 (2007) 367–380.
[5] Y. Zhang, W.E. Leithead, D.J. Leith, Time-series Gaussian process regression based on toeplitz computation of $O(N^2)$ operations and $O(N)$-level storage, in: Proceedings of the 44th IEEE Conference on Decision and Control, 2005, pp. 3711–3716.

[6] T. Sarkar, K. Siarkiewicz, R. Stratton, Survey of numerical methods for solution of large systems of linear equations for electromagnetic field problems, IEEE Transactions on Antennas and Propagation 29 (1981) 847–856.
[7] R.H. Sturges Jr., Analog matrix inversion (robot kinematics), IEEE Journal of Robotics and Automation 4 (1988) 157–162.
[8] Y. Zhang, Revisit the analog computer and gradient-based neural system for matrix inversion, in: Proceedings of IEEE International Symposium on Intelligent Control, 2005, pp. 1411–1416.
[9] K.S. Yeung, F. Kumbi, Symbolic matrix inversion with application to electronic circuits, IEEE Transactions on Circuits and Systems 35 (1988) 235–238.
[10] A. El-Amawy, A systolic architecture for fast dense matrix inversion, IEEE Transactions on Computers 38 (1989) 449–455.
[11] V.E. Neagoe, Inversion of the Van der Monde matrix, IEEE Signal Processing Letters 3 (1996) 119–120.
[12] Y.Q. Wang, H.B. Gooi, New ordering methods for sparse matrix inversion via diagonalization, IEEE Transactions on Power Systems 12 (1997) 1298–1305.
[13] B. Fathi Vajargah, Different stochastic algorithms to obtain matrix inversion, Applied Mathematics and Computation 189 (2007) 1841–1846.
[14] C.K. Koc, G. Chen, Inversion of all principal submatrices of a matrix, IEEE Transactions on Aerospace and Electronic Systems 30 (1994) 280–281.
[15] R.K. Manherz, B.W. Jordan, S.L. Hakimi, Analog methods for computation of the generalized inverse, IEEE Transactions on Automatic Control 13 (1968) 582–585.
[16] J. Jang, S. Lee, S. Shin, An optimization network for matrix inversion, Neural Information Processing Systems, American Institute of Physics, New York, 1988.
[17] J. Wang, A recurrent neural network for real-time matrix inversion, Applied Mathematics and Computation 55 (1993) 89–100.
[18] Y. Zhang, D. Jiang, J. Wang, A recurrent neural network for solving Sylvester equation with time-varying coefficients, IEEE Transactions on Neural Networks 13 (2002) 1053–1063.
[19] Y. Zhang, S.S. Ge, A general recurrent neural network model for time-varying matrix inversion, in: Proceedings of the 42nd IEEE Conference on Decision and Control, 2003, pp. 6169–6174.
[20] Y. Zhang, S.S. Ge, Design and analysis of a general recurrent neural network model for time-varying matrix inversion, IEEE Transactions on Neural Networks 16 (2005) 1477–1490.
[21] N.C.F. Carneiro, L.P. Caloba, A new algorithm for analog matrix inversion, in: Proceedings of the 38th Midwest Symposium on Circuits and Systems, 1995, pp. 401–404.
[22] C. Mead, Analog VLSI and Neural Systems, Addison-Wesley, Reading, MA, 1989.
[23] Y. Zhang, J. Wang, Global exponential stability of recurrent neural networks for synthesizing linear feedback control systems via pole assignment, IEEE Transactions on Neural Networks 13 (2002) 633–644.
[24] Y. Zhang, A set of nonlinear equations and inequalities arsing in robotics and its online solution via a primal neural network, Neurocomputing 70 (2006) 513–524.
[25] The MathWorks, Inc.: Matlab 7.0, Natick, MA, 2004.