第 16 卷 第 4 期
1994 年 7 月

电子科学学刊
JOURNAL OF ELECTRONICS

Vol. 16, No. 4
July, 1994

求解多项式实零点问题的神经网络法*

张青富　　保　铮
（西安电子科技大学电子工程研究所　西安　710071）

摘要　本文给出一类适合于求解多项式实零点问题的神经网络. 理论分析和模拟结果都表明,这类网络可实时求解多项式实零点问题.

关键词　神经网络,多项式,信号处理

1 引　言

多项式零点问题是数值计算中最基本的问题之一，也是在工程技术领域经常遇到的一个问题. 在信号处理、自动控制等领域，人们常需要实时求解这一问题. 传统的算法，如 Newton 类算法[1]和近年来出现的新算法，如 Kuhn 算法[2]都很难实时求解多项式零点问题.

近年来,人们越来越重视人工神经网络的计算功能,神经网络在优化问题及其相关问题上已显示了并行集体实时计算能力[3-9]. 事实上,目前绝大多数这方面的研究是针对优化问题或容易转化为优化问题的问题. 本文将给出一类适合于求解多项式实零点问题的神经网络模型.

2 一类神经网络及其动态分析

我们考察如下神经网络模型:

$$dV/dt = -AV + \beta(V^TDV)^\alpha V, \tag{1}$$

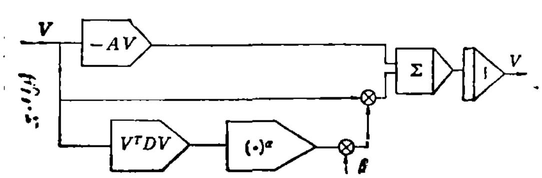

图　1

这里 $A=(a_{ij})_{n\times n}\in R^{n\times n}$, A的特征值都是正实数; $D=(d_{ij})_{n\times n}$ 是 $n\times n$ 阶实对称正定矩阵; $V=(V_1,\cdots,V_n)^T\in R^n$; $\alpha<0$; $\beta>0$.

实现(1)式的电路框图如图 1 所示. 由常微分方程理论[10]易证:

定理 1　对于任意非零的 $V_0=(V_{01},\cdots,V_{0n})^T\in R^n$,满足 $V(0)=V_0$ 的(1)式的解

1993-02-16 收到,1993-10-28 定稿
* 国家自然科学基金资助项目
张青富　男, 1965 年生,博士研究生,现从事神经网络和最优化理论在信号处理中的应用方面的研究工作.
保　铮　男, 1927 年生,中国科学院院士,教授,博士生导师,现从事雷达系统和信号处理方面的研究工作.

唯一存在。

由于A的特征值都是正实数，根据线性代数有关理论[11]，对于任意实数 $\varepsilon \neq 0$，存在矩阵 $T \in R^{n\times n}$，使

$$T^{-1}AT = \mathrm{diag}[J_1(\lambda_1), J_2(\lambda_2), \cdots, J_k(\lambda_k)]. \tag{2}$$

这里 $0 < \lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_k$，$J_i(\lambda_i)$ 的阶数是 r_i，$\sum_{i=1}^{k} r_i = n$，且

$$J_i(\lambda_i) = \begin{pmatrix} \lambda_i & \varepsilon & & & 0 \\ & \lambda_i & \varepsilon & & \\ & & \lambda_i & \ddots & \\ & & & \ddots & \varepsilon \\ 0 & & & & \lambda_i \end{pmatrix}. \tag{3}$$

在变换 $V = TU$ 作用下，(1)式等价于

$$\mathrm{d}U/\mathrm{d}t = -T^{-1}ATU + \beta(U^T T^T DTU)^\alpha U. \tag{4}$$

注意到 $T^T DT$ 仍是正定对称矩阵，所以，在以下对(1)式的讨论分析中，我们总假设

$$A = \mathrm{diag}[J_1(\lambda_1), J_2(\lambda_2), \cdots, J_k(\lambda_k)]. \tag{5}$$

由于在(3)式中 ε 的任意性，以下我们总假设

$$0 < \varepsilon < \lambda_1/8. \tag{6}$$

定理 2　设 $V_0 \in R^n$，$V_0 \neq 0$，$V = V(t, V_0)$ 是(1)式满足 $V(0) = V_0$ 的解，则存在 $M_1(V_0) > 0$，$M_2(V_0) > 0$，使

$$M_1(V_0) \leqslant \|V(t, V_0)\| \leqslant M_2(V_0). \tag{7}$$

证明　$$(1/2)\mathrm{d}\|V\|^2/\mathrm{d}t = -V^T AV + \beta(V^T DV)^\alpha V^T V. \tag{8}$$

因为(5)式、(6)式，$V^T AV$ 是正定二次型，这样就存在常数 $L_1 > 0, L_2 > 0$ 使

当 $\|V\| > L_1$ 时，

$$-V^T AV + \beta(V^T DV)^\alpha V^T V < 0; \tag{9}$$

当 $\|V\| < L_2$ 时，

$$-V^T AV + \beta(V^T DV)^\alpha V^T V > 0. \tag{10}$$

由(9)式、(10)式，可知本定理成立.　　证毕

由定理 2，神经网络模型(1)式的动态范围是有界的，这对于这一模型的硬件实现是有意义的.

根据常微分方程组解的延拓定理[10]，可证

定理 3　$\forall V_0 \in R^n$，$V_0 \neq 0$，(1)式的解 $V = V(t, V_0)$ 可唯一地延拓到$[0, +\infty)$上.

下面研究(1)式的稳定性.

定理 4　(1)如果 $\bar{V}$ 是(1)式的平衡点，则 $\bar{V}$ 是A的对应于特征值 $\beta(\bar{V}^T D\bar{V})^\alpha$ 的特征向量.(2)如果 $\bar{X}$ 是A的对应于特征值 λ 的特征向量，则 $\pm\lambda^{1/(2\alpha)}\beta^{-1/(2\alpha)}(\bar{X}^T D\bar{X})^{-1/2}\bar{X}$ 是(1)式的平衡点.

证明　(1)显然成立.(2)如果 $\bar{X}$ 是(1)式对应于特征值 λ 的特征向量，则

$$A\bar{X} = \lambda\bar{X},\ \text{且}\ \bar{X} \neq 0.$$

记　$r = \lambda^{1/(2\alpha)}\beta^{-1/(2\alpha)}(\bar{X}^T D\bar{X})^{-1/2}$，我们有

$$\begin{aligned} &-A(\pm r\bar{X}) + \beta[(\pm r\bar{X})^T D(\pm r\bar{X})]^\alpha(\pm r\bar{X}) \\ &= \pm r[-A\bar{X} + \beta r^{2\alpha}(\bar{X}^T D\bar{X})^\alpha\bar{X}] \end{aligned}$$

$$= \pm[-A\bar{X} + \lambda\bar{X}] = 0.$$

所以$\pm r\bar{X}$ 是(1)式的平衡点. 证毕

从计算的角度出发,我们希望神经网络模型(1)式具有稳定点.

定理5 (1)如果 $\lambda_1 < \lambda_2$, 且 $r_i = 1$,即 λ_1 是单特征根, $0 < \varepsilon < (\lambda_2 - \lambda_1)/8$, 则 $V^* = (a, 0, \cdots, 0)^T$ 和$-V^*$是(1)式的指数稳定点,这里 $a = d_{11}^{-1/2}\lambda_1^{1/(2\alpha)}\beta^{-1/(2\alpha)}$.

(2) 如果 $\lambda_1 < \lambda_i$,即 λ_i 不是A的最小特征值,则对应于 λ_i 的(1)式的平衡点 $\bar{V}$ 是不稳定点.

证明 (1)我们只讨论 V^* 情形,$-V^*$ 情形同理可证. 我们先证明如下事实.

设 $\Delta V = V - V^*, \Delta V = (\Delta V_1, \cdots, \Delta V_n)^T$,则存在常数 $1 > r > 0$,当 $0 < |\Delta V_1| < a$, 且 $\left[\sum_{i=2}^{n} (\Delta V_i)^2\right]^{1/2} \leqslant r|\Delta V_1|$ 时, $\beta(V^TDV)^\alpha - \lambda_1$ 与 ΔV_1 异号, 此时 $dV_1/dt = [\beta(V^TDV)^\alpha - \lambda_1]V_1$ 与 ΔV_1 异号.

设 $\overline{\Delta V} = \Delta V/\Delta V_1$, 当 $0 < |\Delta V_1| < a$ 时

$$\begin{aligned} V^TDV &= V^{*T}DV^* + 2V^{*T}D\Delta V + \Delta V^TD\Delta V \\ &= (\lambda_1/\beta)^{1/\alpha} + \Delta V_1(2V^{*T}D\overline{\Delta V} + \Delta V_1\overline{\Delta V}^TD\overline{\Delta V}), \end{aligned} \tag{11}$$

$$\begin{aligned} &2V^{*T}D\overline{\Delta V} + \Delta V_1\overline{\Delta V}^TD\overline{\Delta V} \\ &= 2ad_{11} + \Delta V_1 d_{11} + O\left(\left[\sum_{i=2}^{n}(\Delta V_i)^2\right]^{1/2} \Big/ \Delta V_1\right) \\ &= (2a + \Delta V_1)d_{11} + O\left(\left[\sum_{i=2}^{n}(\Delta V_i)^2\right]^{1/2} \Big/ \Delta V_1\right). \end{aligned} \tag{12}$$

由(11)式、(12)式知 r 的存在性.

我们取 V^* 的邻域 $N(V^*, \varepsilon_0)$,使其满足下列条件,即

当 $V \in N(V^*, 2\varepsilon_0/r)$ 时,

$$\lambda_2 - \beta(V^TDV)^\alpha \geqslant (\lambda_2 - \lambda_1)/2 = k, \tag{13}$$

$$\varepsilon_0 < ra/4. \tag{14}$$

下面,我们考察(1)式的解 $V = V(t, V_0)$,这里 $V_0 \in N(V^*, \varepsilon_0)$. 记

$$\mathrm{diag}[J_2(\lambda_2 - \beta(V^TDV)^\alpha), \cdots, J_k(\lambda_k - \beta(V^TDV)^\alpha)] = B(V). \tag{15}$$

由于 $0 < \varepsilon < (\lambda_2 - \lambda_1)/8$,再注意到(13)式可知,当 $V \in N(V^*, 2\varepsilon_0/r)$ 时,存在常数 $f > 0$,使 $\{B(V) + [B(V)]^T\}/2$ 的特征值都大于 f. 此时,

$$\begin{aligned} d\left(\sum_{i=2}^{n} V_i^2\right) \Big/ dt &= 2\sum_{i=2}^{n} V_i \frac{dV_i}{dt} \\ &= -(V_2, \cdots, V_n)[B(V) + B^T(V)](V_2, \cdots, V_n)^T \\ &\leqslant -2f\sum_{i=2}^{n} V_i^2. \end{aligned} \tag{16}$$

这样当 $V = V(t, V_0) \in N(V^*, 2\varepsilon_0/r)$ 时,

$$\sum_{i=2}^{n} V_i^2 \leqslant \left(\sum_{i=2}^{n} V_{0i}^2\right)\exp(-2ft). \tag{17}$$

又因为当 $V = V(t,V_0)\in N(V^*, 2\varepsilon_0/r)$，$\left(\sum_{i=2}^{n} V_i^2\right)^{1/2} \leqslant r|V_1 - a|$ 时，dV_1/dt 与 $V_1 - a$ 异号，且 $V_0\in N(V^*,\varepsilon_0)$，所以

$$|V_1 - a| \leqslant (1/r)\|V^* - V_0\|\exp(-ft). \tag{18}$$

由(17)式、(18)式，我们有

$$\|V - V^*\| \leqslant (1 + 1/r)\|V^* - V_0\|\exp(-ft). \tag{19}$$

由以上讨论可知，对于 $V = V(t,V_0)$，当 $V_0\in N(V^*,\varepsilon_0)$时，对于任意 $t > 0$，都有

$$\|V - V^*\| \leqslant (2/r)\|V^* - V_0\|\exp(-ft). \tag{20}$$

这就证明了 V^* 是(1)式的指数稳定点.

(2) 由于 $\beta(\bar{V}^T D\bar{V})^\alpha = \lambda_i$，所以，存在 $\bar{V}$ 的一个邻域 $N(\bar{V},\varepsilon_1)$，使当 $V\in N(\bar{V},\varepsilon_1)$ 时，

$$\beta(\bar{V}^T D\bar{V})^\alpha - \lambda_1 \geqslant (\lambda_i - \lambda_1)/2 = g. \tag{21}$$

这样对于任意的正数 $\delta < \varepsilon_1$，取 $V_0 = (V_{01},\cdots,V_{0n})^T\in N(\bar{V},\delta)$，使 $V_{01}\neq 0$，则当(1)式的解 $V = V(t,V_0)\in N(\bar{V},\varepsilon_1)$时，

$$\begin{aligned} &dV_1/dt = [\beta(V^T DV)^\alpha - \lambda_1]V_1, \\ &dV_1^2/dt \geqslant 2gV_1^2, \\ &V_1^2 \geqslant V_{01}^2\exp(2gt), \\ &|V_1| \geqslant |V_{01}|\exp(gt). \end{aligned} \tag{22}$$

由(22)式知，存在 $t_0 > 0$，使

$$\|V(t_0,V_0) - \bar{V}\| \geqslant \varepsilon_1. \tag{23}$$

这就证明了 $\bar{V}$ 是不稳定点.　　证毕

由以上讨论可知，当 A 的最小特征值是正的单特征值时，神经网络模型(1)式的任一解都是有界的，而且(1)式具有指数稳定点，在稳定点处，$\beta(V^T DV)^\alpha$ 一定是 A 的最小特征值.

3　基于神经网络(1)式的求解多项式实零点问题的方法

我们考虑多项式

$$x^n + a_{n-1}x^{n-1} + \cdots + a_1x + a_0 = 0 \tag{24}$$

的零点问题，这里 $a_i\in R(i = 0,\cdots,n - 1)$. 假设(24)式的零点都是正实数.

为利用神经网络(1)式求解(24)式，我们选择

$$A = \begin{pmatrix} 0, & 1, & & \cdots, & 0 \\ 0, & 0, & 1, & \cdots, & 0 \\ \cdots & \cdots & & & \\ -a_0, & -a_1, & & \cdots, & -a_{n-1} \end{pmatrix}. \tag{25}$$

容易验证，A 的特征多项式就是(24)式. 这样当(24)式的最小零点是单根时，由上节的讨论可知，利用神经网络模型(1)式可求解多项式(24)式的零点问题.

如果(24)式的零点是实数，其最小零点是负数，我们可选择

$$A=\begin{pmatrix} M, & 1, & \cdots, & 0 \\ 0, & M, & 1, \cdots, & 0 \\ -a_0, & -a_1, & \cdots, & M-a_{n-1} \end{pmatrix}. \tag{26}$$

这里M是一足够大的正数,A的特征值都是正数。易知,如果λ是A的最小特征值,则$\lambda-M$是(24)式的最小零点。这样神经网络(1)式可用来求(24)式的负的最小零点。

如果需要求(24)式的最大零点,我们可选择

$$A=\begin{pmatrix} \bar{M}, & -1, & & \cdots, & 0 \\ 0, & \bar{M}, & -1, & \cdots, & 0 \\ \cdots & & \cdots & & \\ a_0, & a_1, & \cdots & & \bar{M}+a_{n-1} \end{pmatrix}. \tag{27}$$

这里$\bar{M}$是一足够大的正数,使A的特征值都大于零。易知,如果λ是A的最小特征值,则$\bar{M}-\lambda$是(24)式的最大零点。

在一些情况下,我们需要求出(24)式的多个零点或全部零点。当我们求出(24)式的第一个零点λ_1后,利用如下的算法

$$\left.\begin{aligned} b_{n-2} &= \lambda_1 + a_{n-2}, \\ b_{n-i} &= \lambda_1 b_{n+1-i} + a_{n-i},\ i=3,4,\cdots,n \end{aligned}\right\} \tag{28}$$

递归计算出 $b_{n-2},\cdots,b_0$,则

$$x^{n-1}+b_{n-2}x^{n-2}+\cdots+b_0=0 \tag{29}$$

的$(n-1)$个零点是(24)式的除λ_1外的$(n-1)$个零点。

对(29)式利用本文的神经网络方法求零点,可求出(24)式的第二个零点。如此递归下去,我们可求出(24)式的全部实零点。

应该指出,在利用神经网络(1)式求解多项式实零点问题时,如果参数矩阵D存在误差ΔD,那么,只要$D+(\Delta D+\Delta D^T)/2$正定,则误差$\Delta D$不影响求解。

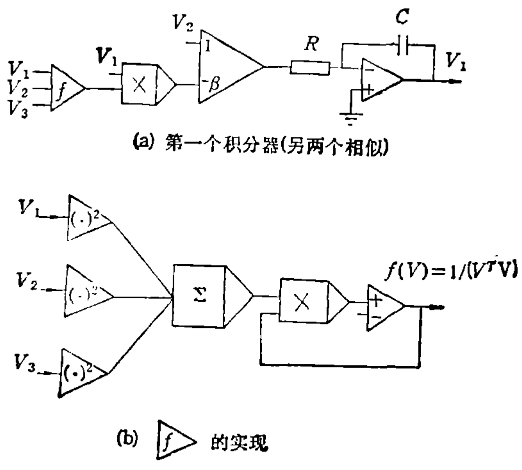

图 2

4 模拟结果

我们模拟了本文提出的求解多项式实零点问题的神经网络方法。

例 1 求多项式 $x^3-2.5x^2+2x-0.5=0$ 的零点。

其最小零点是 0.5。

根据本文方法,我们给出了如图 2 所示的求解此问题的神经网络电路。这里,我们使用了三个相同的 RC 积分器,RC 时间常数决定了积分速度,从而也决定了收敛速度,我们取 $R=1\text{k}\Omega$,$C=10$ nF。

描述此电路的动态方程为

$$dV_1/dt = [1/(RC)](-V_2 + \beta V_1/\|V\|^2),$$
$$dV_2/dt = [1/(RC)](-V_3 + \beta V_2/\|V\|^2),$$
$$dV_3/dt = [1/(RC)](-0.5V_1 + 2V_2 - 2.5V_3 + \beta V_3/\|V\|^2).$$

这里我们取D是三阶单位阵，$V = (V_1,V_2,V_3)^T$.

我们对不同的β值和不同的初始条件做了模拟试验，部分结果如下：

(1)　$\beta = 7$, $V(0) = (1,0,0)^T$;

模拟结果　$x = 0.5000008$.

(2)　$\beta = 60$, $V(0) = (10,9,3)^T$;

模拟结果　$x = 0.5000008$.

(3)　$\beta = 0.6$, $V(0) = (0,7,0)^T$;

模拟结果　$x = 0.5000007$.

例 2　求多项式 $x^5 - 4.4x^4 + 7.6x^3 - 6.4x^2 + 2.6x - 0.4 = 0$ 的零点.

与求解例 1 相同，我们可给出求解这一问题的神经网络电路，我们仍取 $R = 1\mathrm{k}\Omega$, $C = 10\mathrm{nF}$.

部分模拟结果如下：

(1)　$\beta = 6$, $V(0) = (1,0,0,0,0)^T$;

模拟结果　$x = 3.999994 \times 10^{-1}$.

(2)　$\beta = 20$, $V(0) = (0,0,0,3,9)^T$;

模拟结果　$x = 3.999996 \times 10^{-1}$.

(3)　$\beta = 16$, $V(0) = (2,0,-5,0,29)^T$;

模拟结果　$x = 3.999994 \times 10^{-1}$.

在以上两例试验中，模拟时间均是 100μs.

上述模拟结果表明，本文给出的神经网络方法可实时求解多项式零点问题.

5 结 束 语

本文给出了一类神经网络计算模型，理论分析和模拟都表明，这类网络可用来实时求解多项式实零点问题. 因此，在实时信号处理等工程领域，这一神经网络计算模型具有实用价值.

参 考 文 献

[1] Stoer J, Bulirsch R. Introduction to Numerical Analysis, New York: Spring-Verlag, 1980, 270—290.

[2] 陈开周，党创寅，杨再福. 不动点理论和算法. 西安：西安电子科技大学出版社，1990，1—18.

[3] Tank D W, Hopfield J J. IEEE Trans. on CAS, 1986, CAS-33(5): 533—541.

[4] Kennedy M P, Chua L O. IEEE Trans. on CAS, 1988, CAS-35(5): 554—562.

[5] Zhang S, Zhu X, Zuo L H. IEEE Trans. on NN, 1992, NN-3(6): 1021—1024.

[6] Zhuang X, Zhao Y, Huang T S. Neural Networks, 1991, 4(3): 619—626.

[7] Ingman D, Merlis Y. IEEE Trans. on NN, 1992, NN-3(2): 195—201.

[8] Wang J. Pattern Recognition and Artificial Intelligence, 1991, 5(4): 581—601.

[9] Osowski S. Int. J. Circuit Theory and Application, 1992, 20(1):93—98.

[10] Miller R K, Michel A N. Ordinary Differential Equation. New York: Academic Press, 39—63, 167—168.
[11] 谢邦杰. 线性代数. 北京: 人民教育出版社,1978,171—201.

NEURAL NETWORK ALGORITHM FOR COMPUTING REAL ZEROS OF POLYNOMIALS

Zhang Qinfu Bao Zheng
(Institute of Electronics Engineering, Xidian University, Xi'an 710071)

Abstract A neural network model for computing real zeros of polynomials is presented. Both the mathematical analysis and the experimental results show that the proposed network is effective.

Key words Neural network, Polynomials, Signal processing