

神经网络 PID 控制器在硬盘磁头定位系统中的应用

张洪波, 黄心汉, 彭刚

ZHANG Hong-bo, HUANG Xin-han, PENG Gang

华中科技大学 控制科学与工程系, 武汉 430074

Department of Control Science & Engineering, Huazhong University of Science and Technology, Wuhan 430074, China

E-mail: zg_hongbo@sina.com

ZHANG Hong-bo, HUANG Xin-han, PENG Gang. Application of neural network PID controller in HDD magnetic head positioning system. Computer Engineering and Applications, 2007, 43(25): 89-92.

Abstract: In Hard Disk Drives(HDD), it is hard to establish accurate plant model. Aimed at this characteristic, this paper presents a method which applies neural network combined with PID controller. This scheme can realize real-time and on-line adjustment for proportional, integral and derivative parameters of PID controller using the capabilities of self-learning and arbitrary non-linear expressions of neural network, as a result, contributes to finding the optimal combination of PID parameters so as to attain optimization of some performance target. As the modified style of conventional BP neural network, this paper uses the rule of minimizing total square error from initial time to current time as the performance function. The simulation results show that, applying this control scheme for HDD servo control, rapid convergent speed, short settling time, almost no overshoot as well as no steady-state error can be obtained. This method has superior to conventional one which is suitable for actual HDD magnetic head positioning.

Key words: Neural Network; PID; Hard Disk Drive; magnetic head positioning

摘要: 针对硬盘驱动器难以建立准确对象模型的特性, 提出了一种采用神经网络 PID 控制器的方法。该方法利用神经网络的自学习能力和任意非线性表达能力, 实现实时、在线地调整 PID 控制器的比例、积分、微分系数, 从而找到 PID 控制参数的最佳组合, 以达到某种性能指标的最优化。作为常规 BP 神经网络的改进型, 提出以从最初时刻到当前时刻的误差的平方和最小作为性能指标函数。仿真结果表明, 采用这种控制方案进行硬盘伺服控制, 系统收敛速度快、调节时间短、且几乎没有超调和稳态误差, 性能优于常规神经网络 PID 控制, 适用于实际的硬盘驱动器磁头定位。

关键词: 神经网络; PID; 硬盘驱动器; 磁头定位

文章编号: 1002-8331(2007)25-0089-04 **文献标识码:** A **中图分类号:** TP273; TP333

1 引言

硬盘驱动器作为一种存储数据信息的设备, 在目前的计算机系统中起着不可替代的作用。众所周知, 第一块硬盘驱动器由 IBM 公司于 1956 年研制出来, 当时容量仅为 4.4 MB。随着技术的进步, 在短短的半个世纪内, 硬盘驱动器的最大容量已达到 750 GB。硬盘驱动器容量的增大在很大程度上归功于准确的伺服定位机构。在目前的伺服定位算法中, 比较典型的有由 S.M.Suh^[1]采用的 LQG/LTR 技术, Sung-Won Park^[2]提出的重复控制器(Repetitive Controller), Young-Man Choi^[3]提出的 PTOS (Proximate Time Optimal Servomechanism) 技术, 以及 Muhammad A.Hawwa^[4]采用的非线性 PID 控制方法等。这些方法在获取准确的对象模型条件下都可以获得较好的控制效果。

如今硬盘驱动器正朝着更大容量、更小尺寸的方向发展, 这两种需求一方面势必使磁道间距越来越窄, 另一方面使系统

的非线性性和不确定性也随之加大, 要想获得好的控制效果, 需要更为准确、更为鲁棒的磁头定位算法。如此以来, 随着磁道间距的减小, 以往由枢轴摩擦、主轴倾斜、盘片滑动、偏移力和一些扰动等诸多因素所产生的影响不能再被忽略。然而由于摩擦等因素存在高度非线性^[5], 迄今还没有好的方法能建立其准确的数学模型。这些非线性性和不确定性直接导致建立准确的对象模型难度也随之加大, 有时甚至是不可能的, 因而在这种情况下有必要研究不依赖对象模型的控制算法。本文的神经网络就是在这样的背景下提出来的, 神经网络^[6]的学习能力及可以逼近任意非线性函数的能力使得它在控制非线性被控系统中有着天然的优势。另外 PID 控制由于算法简单, 实现起来也较容易, 这使得它自问世以来在工业过程控制中倍受青睐。然而最优的比例、积分、微分系数很难确定, 这成为限制它在复杂被控系统中应用的瓶颈。因而单纯的 PID 控制很难在非线性

作者简介: 张洪波(1978-), 男, 博士生, 研究方向: 智能控制、硬盘驱动器磁头定位等; 黄心汉(1946-), 男, 教授, 博士生导师, 研究方向: 智能控制、机器人、智能传感器、多传感器集成与信息融合等; 彭刚(1973-), 男, 副教授, 研究方向: 移动机器人、人工智能、嵌入式系统与软件开发、计算机工业控制等。

系统中获得好的控制效果。将 PID 控制和其他先进控制方法结合起来成为必然^[7-9]。神经网络的自学习能力可以很好地做到在线调整 PID 的三个参数。因而本文尝试着将神经网络和 PID 控制结合起来实现硬盘磁头定位,利用神经网络实时调整 PID 参数,以达到自适应地跟踪参考输入的目的,进而实现快速、无静差的磁头定位。

为验证设计方法的可行性,给出了仿真结果,其结果表明该设计方案是行之有效的,适用于实际的硬盘驱动器伺服控制系统。

2 神经网络 PID 控制算法

神经网络 PID 控制(Neural Network PID, NNPID)的结构如图 1 所示,采用通常的三层结构,即输入层、隐含层和输出层。输入层和隐含层节点的数目可以根据被控对象的复杂度自行选择,输出层有三个节点,分别对应 PID 控制的三个可调参数。为了减少权值更新次数,加快网络收敛速度,以便尽快找到 PID 参数的最佳非线性组合,从而使选定的性能指标达到最优,可以事先选定一组可选的 PID 参数(K'_p, K'_I, K'_D),从而 PID 参数的调整限制在一个较小的范围内进行。

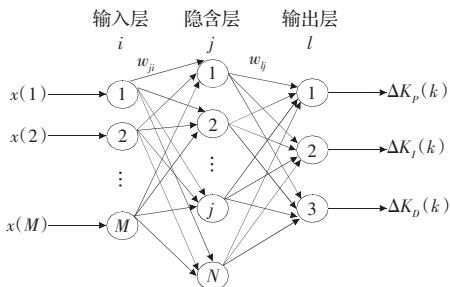


图 1 神经网络结构图

值得注意的是,常规的 BP 神经网络,主要是以最小二乘(Least Mean Square, LMS)算法作为学习算法^[10,11],其思想是在网络学习过程中,以当前的系统误差平方最小作为性能指标从而调整网络加权系数,这没有考虑到以前的误差对当前网络加权系数调整的影响。因此本文尝试着以从最初时刻到当前时刻的误差的平方和最小作为代价函数,这样做充分考虑到过去的误差对当前权值调整的影响,另外考虑到越旧的误差对当前加权系数调整的影响越小,因此本文引入一个 λ 因子($0 \leq \lambda < 1$),即遗忘因子。具体来说就是使系统性能指标由如下的式(1)变为式(2):

$$J(k) = \frac{1}{2} (r(k) - y(k))^2 \tag{1}$$

$$J(k) = \frac{1}{2} \sum_{n=1}^k \lambda^{k-n} (r(n) - y(n))^2 \tag{2}$$

式(2)中 $r(n)$ 表示第 n 时刻系统的参考输入, $y(n)$ 表示第 n 时刻系统的实际输出, k 表示当前时刻。

令系统误差 $e(n) = r(n) - y(n)$, 则第 k 时刻增量式 PID 控制器的算法可写为如下形式

$$u(k) = u(k-1) + K_p(k)[e(k) - e(k-1)] + K_I(k)e(k) + K_D(k)[e(k) - 2e(k-1) + e(k+2)] \tag{3}$$

式中 $K_p(k), K_I(k), K_D(k)$ 分别对应 PID 控制器的比例、积分、微分系数。

神经网络的自学习过程分为两步:前向传播过程和误差反

向传播过程^[12]。其中前向传播过程完成从输入层到输出层的非线性映射过程,进而产生作用于被控对象的控制量,以实时控制系统输出,将其与参考输入进行比较后产生系统误差;误差反向传播过程主要是神经网络将系统误差进行反向传播,进而调整各网络层加权系数,使系统误差逐步朝着减小的方向演进。

下面推导出以式(2)作为性能指标的神经网络加权系数更新公式。

2.1 前向传播过程

该过程同常规 BP 神经网络^[10,12]。由图 1 可得:

(1) 输入层

$$\text{输入层输入 } O_i^{(1)}(k) = x(i) (i=1, 2, \dots, M) \tag{4}$$

式(4)中 M 表示输入层神经元节点数, $x(i)$ 为给定的网络输入,其值可由设计人员自行选择,一般选为给定输入、系统输出、系统误差和控制量的某种组合。为简单起见,这里暂不考虑神经元阈值。

(2) 隐含层

隐含层输入、输出可表示为如下形式

$$\begin{cases} net_j^{(2)}(k) = \sum_{i=1}^M w_{ji}^{(2)}(k) O_i^{(1)}(k) \\ O_j^{(2)}(k) = f(net_j^{(2)}(k)) \end{cases} \tag{5}$$

式(5)中 $w_{ji}^{(2)}(k)$ 为输入层和隐含层之间的连接加权系数,根据对象特性本文将隐含层神经元激活函数 $f(\cdot)$ 取为 $f(x) = \frac{b}{1+e^{-ax}}$, 其中 a, b 参数可根据实际被控对象进行选择。

(3) 输出层

网络输出层的输入、输出可写为如下形式

$$\begin{cases} net_l^{(3)}(k) = \sum_{j=1}^N w_{lj}^{(3)}(k) O_j^{(2)}(k) \\ O_l^{(3)}(k) = g(net_l^{(3)}(k)) \end{cases} \tag{6}$$

式(6)中 N 表示隐含层神经元节点数, $w_{lj}^{(3)}(k)$ 为隐含层和输出层之间的连接加权系数,输出层神经元激活函数 $g(\cdot)$ 取为 $g(x) = \frac{d}{1+e^{-cx}}$, 其中 c, d 参数可根据实际被控对象进行选择。

神经网络结构的三个输出分别对应 PID 控制器的三个可调参数,即 $O_1^{(3)}(k) = \Delta P(k), O_2^{(3)}(k) = \Delta I(k), O_3^{(3)}(k) = \Delta D(k)$ 。因此 PID 控制器的系数为 $K_p(k) = K'_p + \Delta P(k), K_I(k) = K'_I + \Delta I(k), K_D(k) = K'_D + \Delta D(k)$ 。

上述各式中的上标值(1)、(2)、(3)分别对应其网络的层数。

2.2 误差反向传播过程

这一过程主要是加权系数的调整过程,具体来说是以 J 对加权系数的负梯度方向作为参数调整方向,以使性能指标尽快趋于最小,同时为了避免网络搜索陷入局部极小,在加权系数调整中引入两个惯性项。具体调整过程如下:

(1) 隐含层和输出层之间的加权系数调整算式

根据上面的分析可知隐含层和输出层之间的权值更新公式可表为如下形式

$$\Delta w_{ij}^{(3)}(k) = -\eta \frac{\partial J}{\partial w_{ij}^{(3)}} + \alpha \Delta w_{ij}^{(3)}(k-1) + \beta \Delta w_{ij}^{(3)}(k-2) \tag{7}$$

式中 η 为学习速率, α, β 为惯性系数。

式(7)中等式右边第一项的偏导数可计算如下:

$$\frac{\partial J}{\partial w_{ij}^{(3)}} = \frac{\partial J}{\partial y(n)} \cdot \frac{\partial y(n)}{\partial u(k)} \cdot \frac{\partial u(k)}{\partial O_l^{(3)}(k)} \cdot \frac{\partial O_l^{(3)}(k)}{\partial net_l^{(3)}(k)} \cdot \frac{\partial net_l^{(3)}(k)}{\partial w_{ij}^{(3)}} =$$

$$- \sum_{n=1}^k \lambda^{k-n} e(n) \frac{\partial y(n)}{\partial u(k)} \cdot \frac{\partial u(k)}{\partial O_l^{(3)}(k)} \cdot g'(net_l^{(3)}(k)) \cdot O_j^{(2)}(k) \quad (8)$$

由于难以建立准确的被控对象模型, 因而偏导数 $\frac{\partial y(n)}{\partial u(n)}$ 未知, 可以用下列符号函数近似取代

$$\frac{\partial y(n)}{\partial u(k)} = \text{sign} \left(\frac{\partial y(n)}{\partial u(k)} \right) \cdot \left| \frac{\partial y(n)}{\partial u(k)} \right| \quad (n=1, 2, \dots, k) \quad (9)$$

另外根据式(3), 可得:

$$\begin{cases} \frac{\partial u(k)}{\partial O_1^{(3)}(k)} = \frac{\partial u(k)}{\partial K_p(k)} \cdot \frac{\partial K_p(k)}{\partial \Delta K_p(k)} = e(k) - e(k-1) \\ \frac{\partial u(k)}{\partial O_2^{(3)}(k)} = \frac{\partial u(k)}{\partial K_i(k)} \cdot \frac{\partial K_i(k)}{\partial \Delta K_i(k)} = e(k) \\ \frac{\partial u(k)}{\partial O_3^{(3)}(k)} = \frac{\partial u(k)}{\partial K_d(k)} \cdot \frac{\partial K_d(k)}{\partial \Delta K_d(k)} = e(k) - 2e(k-1) + e(k-2) \end{cases} \quad (10)$$

由 $g(\cdot)$ 的形式可得其导数为:

$$g'(net_l^{(3)}(k)) = \frac{c}{d} O_l^{(3)}(k) (d - O_l^{(3)}(k)) \quad (11)$$

令 $\sum_{n=1}^k \lambda^{k-n} e(n) \frac{\partial y(n)}{\partial u(k)} = \gamma(k)$, 则式(8)可重写为:

$$\frac{\partial J}{\partial w_{ij}^{(3)}} = -\gamma(k) \cdot \frac{\partial u(k)}{\partial O_l^{(3)}(k)} \times \frac{c}{d} O_l^{(3)}(k) (d - O_l^{(3)}(k)) \cdot O_j^{(2)}(k) \quad (12)$$

因而隐含层和输出层之间加权系数更新公式为:

$$w_{ij}^{(3)}(k) = w_{ij}^{(3)}(k-1) + \eta \delta_j^{(3)} O_j^{(2)}(k) + \alpha \Delta w_{ij}^{(3)}(k-1) + \beta \Delta w_{ij}^{(3)}(k-2) \quad (13)$$

式中 $\delta_j^{(3)} = \gamma(k) \cdot \frac{\partial u(k)}{\partial O_l^{(3)}(k)} \cdot \frac{c}{d} O_l^{(3)}(k) (d - O_l^{(3)}(k))$ 。

(2) 输入层和隐含层之间的加权系数调整算式

与 $w_{ij}^{(3)}(k)$ 的调整情况类似, 输入层和隐含层之间的加权系数更新公式可表为如下算式:

$$\Delta w_{ji}^{(2)}(k) = -\eta \frac{\partial J}{\partial w_{ji}^{(2)}} + \alpha \Delta w_{ji}^{(2)}(k-1) + \beta \Delta w_{ji}^{(2)}(k-2) \quad (14)$$

其中

$$\frac{\partial J}{\partial w_{ji}^{(2)}} = \frac{\partial J}{\partial y(n)} \cdot \frac{\partial y(n)}{\partial u(k)} \cdot \sum_{l=1}^3 \frac{\partial u(k)}{\partial O_l^{(3)}(k)} \cdot \frac{\partial O_l^{(3)}(k)}{\partial net_l^{(3)}(k)} \times$$

$$\frac{\partial net_l^{(3)}(k)}{\partial O_j^{(2)}(k)} \cdot \frac{\partial O_j^{(2)}(k)}{\partial net_j^{(2)}(k)} \cdot \frac{\partial net_j^{(2)}(k)}{\partial w_{ji}^{(2)}} =$$

$$- \sum_{n=1}^k \lambda^{k-n} e(n) \frac{\partial y(n)}{\partial u(k)} \cdot \sum_{l=1}^3 \frac{\partial u(k)}{\partial O_l^{(3)}(k)} \times g'(net_l^{(3)}(k)) \cdot$$

$$w_{ij}^{(3)}(k) \cdot f'(net_j^{(2)}(k)) \cdot O_i^{(1)}(k) =$$

$$- \sum_{n=1}^3 \delta_l^{(3)}(k) \cdot w_{ij}^{(3)}(k) \cdot \frac{a}{b} O_j^{(2)}(k) (b - O_j^{(2)}(k)) \cdot O_i^{(1)}(k) \quad (15)$$

令 $\delta_j^{(2)} = \sum_{l=1}^3 \delta_l^{(3)} \cdot w_{ij}^{(3)}(k) \cdot \frac{a}{b} O_j^{(2)}(k) (b - O_j^{(2)}(k))$, 则输入层和隐含层之间的加权系数调整算式如下

$$w_{ji}^{(2)}(k) = w_{ji}^{(2)}(k-1) + \eta \delta_j^{(2)} O_i^{(1)}(k) + \alpha \Delta w_{ji}^{(2)}(k-1) + \beta \Delta w_{ji}^{(2)}(k-2) \quad (16)$$

3 神经网络 PID 控制器仿真实验

在目前的硬盘驱动器中, 主要靠音圈电机 (Voice Coil Motor, VCM) 驱动磁头臂进行定位以实现准确的读/写操作。神经网络 PID 控制用于硬盘磁头定位的结构如图 2 所示。图中 VCM 表示被控对象, $r(k)$ 表示系统给定输入, $y(k)$ 为系统实际输出, $e(k)$ 为系统误差, $u(k)$ 为作用于被控对象的控制量, 即 PID 控制器的输出。神经网络采用 6-8-3 结构, $\{r(k), r(k-1), y(k), y(k-1), e(k), u(k-1)\}$ 经过规范化处理后作为神经网络的输入, 其输出为 PID 控制器可调参数 $\{\Delta K_p(k), \Delta K_i(k), \Delta K_d(k)\}$ 。

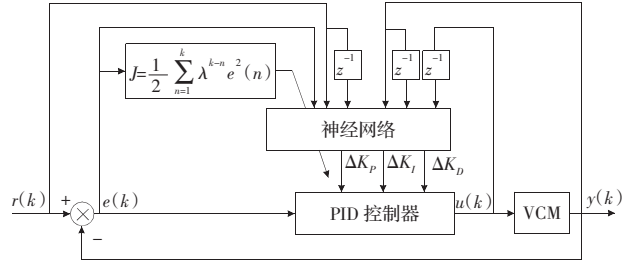


图 2 神经网络 PID 控制结构图

神经网络加权系数的调整是根据使性能指标 J 最小为准则进行的。

根据前面的分析, 事先选择一组可行的 PID 控制器参数, 以使 PID 的比例、积分、微分系数在一个较小的范围内调整。因而整个 PID 控制器参数可表为如下

$$\begin{cases} K_p(k) = 0.2 + \Delta K_p(k) \\ K_i(k) = 0.3 + \Delta K_i(k) \\ K_d(k) = 5.8 + \Delta K_d(k) \end{cases} \quad (17)$$

在该系统中, 选择学习速率 $\eta=0.8$, 惯性系数 $\alpha=0.1, \beta=0.08$, 遗忘因子 $\lambda=0.99$ 。两个激活函数均为 Sigmoid 函数, 其参数分别为 $a=2, b=1.18; c=0.95, d=5.7$ 。

值得注意的是, 恰当选择激活函数也是控制设计的一个主要任务。激活函数的选择对系统性能的影响较大, 选择不当有时甚至无法使系统稳定。作者针对本系统经过多次的研究、实验比较拟选用上述两个激活函数。

PID 控制器采用增量式数字控制器, 其采样频率取为 20 kHz。设给定输入 $r(k)=1 \mu\text{m}$ 。其初始加权系数事先取为 $[-0.5, 0.5]$ 之间的随机值, 待系统稳定后再用稳定加权系数代替该随机值。系统的测试结果如图 3-图 5 所示。

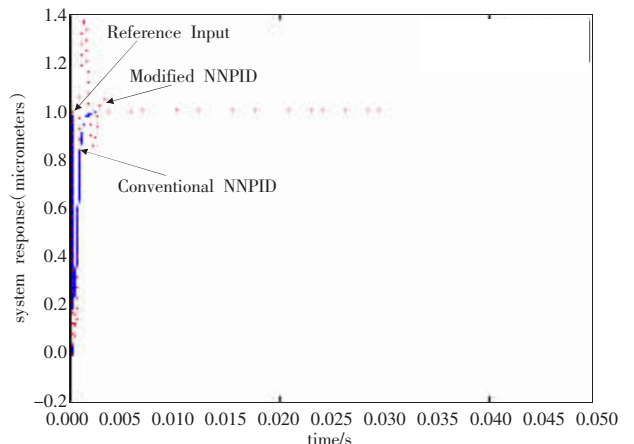


图 3 两种不同性能指标下系统的输出响应曲线

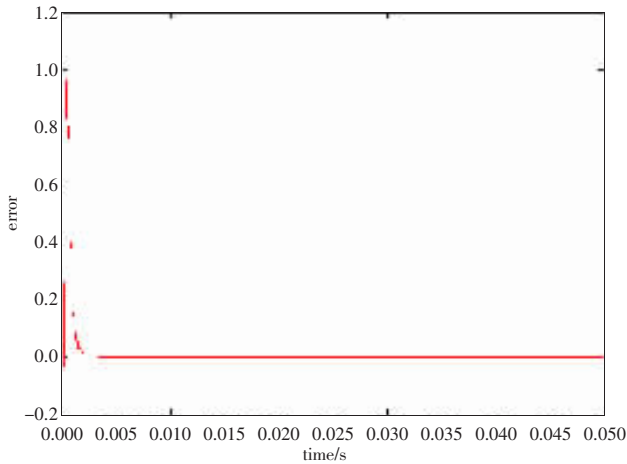


图4 误差曲线

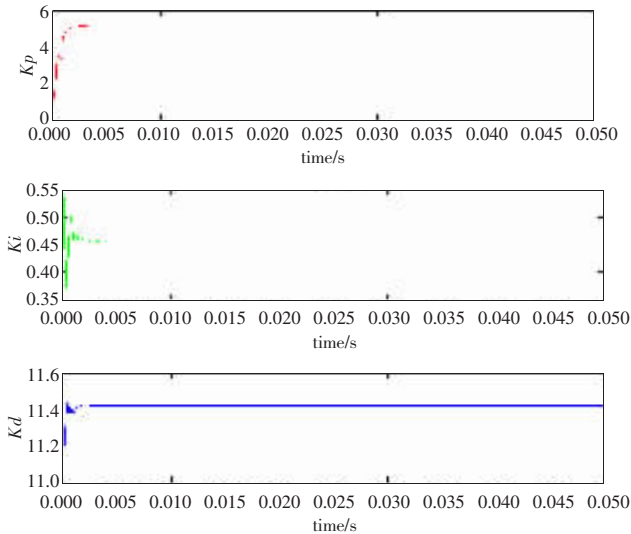


图5 PID参数自适应调整曲线

图3绘出了给定输入和系统实际输出,为便于比较,也给出常规的神经网络PID控制的响应结果。从图中可以看出,本文采用的方法性能更优。图4给出了系统的误差曲线,图5为PID控制器比例、积分、微分系数的自适应调整过程。

表1为在同等条件下从不同方面比较两种算法的性能。其中RMSE为相对均方差,其定义形式如下^[13]:

$$RMSE = 10 \lg \left(\frac{\sum_{k=1}^t (y(k) - r(k))^2}{\sum_{k=1}^t r^2(k)} \right) \quad (18)$$

式中 t 为总迭代次数。

表1 两种不同算法下系统性能比较表

算法	评价指标			
	超调量/%	稳态时间/s	RMSE/dB	稳态迭代次数
常规神经网络PID	36.47	0.006 3	-20.05	125
改进神经网络PID	3.66	0.003 4	-21.33	67

从上述的仿真实验结果和对比表中可以看出采用提出的神经网络PID控制,系统响应快,超调量小,较适合难以建立准

确数学模型的硬盘驱动器磁头定位系统。

4 结束语

针对硬盘驱动器系统中存在的非线性性和不确定性进而难以建立准确数学模型的特性,提出了一种采用神经网络PID控制的方法。在这种方法中,以从最初时刻到当前时刻的误差的平方和最小作为性能指标函数来训练神经网络,利用神经网络的在线学习功能来实时调整PID控制器的参数,以达到PID参数、网络加权系数快速收敛的目的。本文以硬盘驱动器作为被控对象验证了算法的有效性。仿真实验结果表明,改进的智能控制算法收敛速度更快,稳态时间和超调量更小,能实现更为快速、更为准确的磁头定位。(收稿日期:2007年1月)

参考文献:

- [1] Suh S M, Chung C C, Lee S H. Discrete-time LQG/LTR dual-stage controller design in magnetic disk drives[J]. IEEE Trans on Magnetics, 2001, 37(4): 1891-1895.
- [2] Park Sung-Won, Jeong Jun, Yang Hyun Seok, et al. Repetitive controller design for minimum track misregistration in hard disk drives[J]. IEEE Trans on Magnetics, 2005, 41(9): 2522-2528.
- [3] Choi Young-Man, Jeong Jaehwa, Gweon Dae-Gab. A novel damping scheduling scheme for proximate time optimal servomechanisms in hard disk drives[J]. IEEE Trans on Magnetics, 2006, 42(3): 468-472.
- [4] Hawwa M A, Masoud A A. A nonlinear PID servo controller for computer hard disk drives [C]//IEEE International Workshop on Advanced Motion Control, Istanbul, Turkey, 2006: 672-676.
- [5] Herrmann G, Ge Shuzhi Sam, Guo Guoxiao. Practical implementation of a neural network controller in a hard disk drive [J]. IEEE Trans on Control Systems Technology, 2005, 13(1): 146-154.
- [6] Sasaki M, Suzuki T, Ida E, et al. Track-following control of a dual-stage hard disk drive using a neuro-control system[J]. Engineering Applications of Artificial Intelligence, 1998, 11: 707-716.
- [7] Gao Zhenhai, Zhu Bo. Vehicle lane keeping of adaptive PID control with BP neural network self-tuning [C]//IEEE Intelligent Vehicles Symposium, 2005: 84-87.
- [8] Tanaka K, Oka M, Uchibori A, et al. Precise position control of an ultrasonic motor using the PID controller combined with NN[J]. Electrical Engineering in Japan, 2004, 146(3): 46-54.
- [9] Kato M, Yamamoto T, Fujisawa S. A skill-based PID controller using artificial neural networks[C]//Proceedings of the 2005 International Conference on Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, 2005, 1: 702-707.
- [10] 陶永华, 尹怡欣, 葛芦生. 新型PID控制及其应用[M]. 北京: 机械工业出版社, 1998.
- [11] 刘金琨. 先进PID控制及其MATLAB仿真[M]. 北京: 电子工业出版社, 2003.
- [12] Liu Hong-jun, Han Pu, Yao Wan-ye, et al. The research of PID neural network decoupling controller and its application in unit load system[C]//Proceedings of the Third International Conference on Machine Learning and Cybernetics, Shanghai, 2004: 505-509.
- [13] 吴佑寿, 赵明生. 激活函数可调的神经元模型及其有监督学习与应用[J]. 中国科学: E辑, 2001, 31(3): 263-272.