

# 四川 大学

# 硕 士 学 位 论 文

题 目\_\_\_\_\_自适应 LMS 算法的研究及应用\_\_\_\_\_

作 者\_\_\_\_\_马伟富\_\_\_\_\_ 完成日期 \_\_\_\_\_2005 年 5 月\_\_\_\_\_

培养单位\_\_\_\_\_四 川 大 学\_\_\_\_\_

指导教师\_\_\_\_\_雷勇\_\_\_\_\_副教授\_\_\_\_\_

专 业\_\_\_\_\_信号与信息处理\_\_\_\_\_

研究方向\_\_\_\_\_系统监测与网络管理\_\_\_\_\_

授予学位日期\_\_\_\_\_年 月 日\_\_\_\_\_

# 自适应 LMS 算法的研究及应用

信号与信息处理专业

研究生：马伟富 指导教师：雷勇 副教授

现代电子技术已经由模拟向数字过渡，传统的模拟信号处理正被数字信号处理所代替。数字滤波是数字信号处理一部分，它除了具有数字信号处理的稳定、重复性好和适应性强等优点外，还具有可预见性和无相位偏差的优点，可用仿真软件来对一个设计预先测试。实现数字滤波的方法很多，其中，最小均方（LMS）算法在自适应滤波理论中应用最广泛，这是因为 LMS 算法具有低计算复杂度、平稳环境下的收敛性、均值无偏地收敛到维纳解以及利用有限精度实现算法时的稳定性等特性。

本文对几种 LMS 算法进行了仔细研究，包括基本 LMS 算法及其变型、量化误差 LMS 算法、解相关 LMS 算法、并行延时 LMS 算法、自适应格型 LMS 算法和 LMS-Newton 算法，其中量化误差 LMS 算法又包括符号误差 LMS 算法和符号数据 LMS 算法，解相关 LMS 算法包括时域解相关算法和变换域解相关算法。并利用 matlab 对这几种算法进行了软件仿真，通过仿真结果图把各种 LMS 算法的性能直观的展现出来。文中将基本 LMS 算法中不同  $\mu$  时  $e(n)$  的变化绘于一个图形框中，使我们非常直观地看到随着  $\mu$  的增加，算法的学习速率也在逐渐增加，但当  $\mu$  增加到一定程度后，再增加  $\mu$ ，基本 LMS 算法则将陷于发散。同样，文中也给出了其它几种算法的学习速率曲线，并简单地讨论了算法的性能。

此外，文中对 LMS 算法学习速率参数  $\mu$  的选择进行了详细分析，分析了这个参数对 LMS 算法性能产生的影响，以及如何减小影响。文中还对 LMS 算法的统计性能、跟踪性能进行了分析讨论，对算法的失调和非平稳度  $\alpha$  进行了分析，

并对两者之间的关系进行了讨论，给出了采用不同 LMS 算法时的结果。另外，文中还对影响 LMS 算法的几个因素进行了分析说明，分析了这几个因素对 LMS 算法产生什么样的影响，以及产生这些影响的原因。

最后，文中给出了利用 DSP 在数据处理能力和处理速度方面的优势，将 LMS 算法应用到励磁电源控制系统中的实例，对硬件、软件的设计思路进行了说明，给出了硬件实现框图、软件流程图和部分的实际电路和程序。同时还给出了本例中使用 LMS 算法对实际信号进行处理前和处理后的结果。

**关键词：**LMS 算法，自适应滤波，解相关，量化误差，LMS-Newton 算法，DSP

# The Research and Application of Adaptive LMS Algorithm

The Signal and Information Processing

Graduate: Ma Weifu    Adviser: Lei Yong associate professor

College of Electrical Engineering and Information Technology, University Sichuan

Modern electronics technique has already been changed from analog technique toward digital technique, and the traditional analog signal processing has just been replaced by the digital signal processing. The digital filter is a part of digital signal processing, besides having the advantages of the stability, repetition and the good adaptability of the digital signal processing, it has still advantages of predictability and no phase shift, so we can use an imitate software to test a design in advance. We have a lot of methods to carry out digital filter, among them, the application of the least mean square (LMS) algorithm in the theories of adaptive filter is the most extensive. The reason is that the LMS algorithm has characteristics that includes low calculation complications, the constringency under steady environment, the mean no shift constringe to the Wiener solution and the stability using the limited accuracy to carry out the algorithm.

This paper researches sorts of LMS algorithms in detail, including the basic LMS and it's changed type, quantization error LMS algorithm, decorrelation LMS algorithm, adaptive lattic LMS algorithm, parallel delayed LMS algorithm, and LMS - Newton algorithm. Among them, the

quantization error LMS algorithm includes the sign error LMS algorithm and the sign data LMS algorithms, the decorrelation LMS algorithm includes time domain decorrelation LMS algorithm and transform domain LMS algorithm. Also, this paper utilizes MATLAB to imitate these sorts of algorithms, through the simulate result diagram, intuitionistic revelation various characteristics of LMS algorithm. In this paper, we draw the study curve that basic LMS algorithm that  $\mu$  is not same, in a sketch frame, this makes us keep intuitionistic view to see that along with increment of  $\mu$ , the study curve of the algorithm also in the gradual increment. However, when  $\mu$  reaches a certainly value, then, if  $\mu$  is continue increment, the study curve of basic LMS algorithm would fall into divergence. Same as the basic LMS algorithm, this paper also give study curve of the other sorts LMS algorithm, and discussed the performance of the other sorts LMS algorithm in brief.

In addition, this paper detailed analysis the selection method that the study velocity parameter of LMS algorithm, analyzed the influence of parameter  $\mu$  to LMS algorithm's performance, and how to minish the influence. It also analyzed the statistics and tracking performance, disadjustment and nonsmoothness. The relation between disadjustment and nonsmoothness is discussed.

Ultimately, this paper gives the project that application the LMS algorithm into the control system of excitation source with the advantage of DSP, and design thinking of hardware, software elucidation, hardware carries out frame diagram, software flow chart and actual electric circuit and parts of program.

**Keywords:** LMS algorithm, divide correlation, quantization error, LMS-Newton, DSP.

# 目 录

1 绪论 .....	1
1.1 引言 .....	1
1.2 自适应 LMS 算法的发展 .....	1
1.2.1 LMS 算法的历史 .....	1
1.2.2 LMS 算法的现状 .....	2
1.2.3 LMS 算法的发展前景 .....	2
1.3 本论文的主要研究内容和创新之处 .....	3
1.3.1 本论文的主要研究内容 .....	3
1.3.2 本论文的创新之处 .....	3
2 自适应 LMS 算法的研究 .....	4
2.1 概述 .....	4
2.2 LMS 算法及其基本变型 .....	4
2.2.1 基本 LMS 算法 .....	6
2.2.2 归一化 LMS 算法 .....	7
2.2.3 加遗忘因子 LMS 算法 .....	7
2.3 量化误差 LMS 算法 .....	8
2.3.1 符号误差 LMS 算法 .....	9
2.3.2 符号数据 LMS 算法 .....	9
2.4 解相关 LMS 算法 .....	10
2.4.1 时域解相关 LMS 算法原理 .....	10
2.4.2 变换域解相关 LMS 算法 .....	11
2.5 并行延时 LMS 算法 .....	12
2.5.1 延时 LMS 算法 .....	13
2.5.2 并行延时 LMS 算法 .....	13
2.5.3 并行延时 LMS 算法性能分析 .....	14
2.6 自适应格型 LMS 算法 .....	17

2.6.1 对称的格型结构 .....	17
2.6.2 格型滤波器设计准则 .....	18
2.6.3 自适应格型 LMS 算法 .....	19
2.7 NEWTON-LMS 算法 .....	22
2.7.1 算法原理 .....	22
2.7.2 性能分析 .....	23
2.8 小结 .....	25
<b>3 LMS 算法的性能分析 .....</b>	<b>26</b>
3.1 学习速率参数的选择 .....	26
3.1.1 均值收敛的条件 .....	26
3.1.2 均方收敛的条件 .....	28
3.1.3 自适应学习速率参数 .....	29
3.2 LMS 算法的统计性能 .....	32
3.2.1 稳态剩余输出均方误差 .....	32
3.2.2 稳态剩余输出能量 .....	34
3.3 LMS 算法的跟踪性能 .....	35
3.3.1 非平稳度 $\alpha$ .....	36
3.3.2 失调 $M$ .....	37
3.3.3 非平稳度 $\alpha$ 与失调 $M$ 之间的关系 .....	38
3.4 影响 LMS 算法的几个因素 .....	39
3.4.1 不稳定性的影响 .....	39
3.4.2 噪声输入端的信号的影响 .....	39
3.4.3 有限字长的影响 .....	40
3.4.4 系数漂移的影响 .....	42
3.5 小结 .....	42
<b>4 LMS 算法在励磁电源控制系统中的应用 .....</b>	<b>43</b>
4.1 概述 .....	43
4.2 系统硬件设计 .....	43

4.2.1 硬件设计思路 .....	43
4.2.2 TMS320VC5410 简介.....	44
4.2.3 A/D、D/A 转换芯片简介.....	45
4.2.4 系统框图 .....	49
4.2.5 实际电路 .....	49
4.3 系统软件设计.....	51
4.3.1 软件设计思路 .....	51
4.3.2 软件流程 .....	52
4.3.3 部分程序 .....	53
4.4 运行结果.....	58
4.5 小结.....	59
结束语 .....	60
参考文献 .....	61
本文作者在读期间科研成果简介 .....	63
声明 .....	64
致谢 .....	65



# 1 绪论

## 1.1 引言

1965 年 Cooley 和 Tukey 在《Mathematics of Computation》上发表了“An Algorithm for the Machine Computation of Complex Fourier Series”(即“快速傅立叶变换算法”)后,数字信号处理这一学科得到了巨大发展,形成了一套较为完整的理论体系。数字滤波是一种数字信号处理操作,它使我们能从含噪声的数据中提取我们需要的信息。数字滤波不但可以实现信号滤波,还可以对信号进行平滑、预测等工作。

随着数字超大规模集成电路技术的发展,数字滤波技术也得到了飞速发展,其中的自适应滤波技术在许多领域得到了广泛的应用。自适应滤波器能够跟踪和适应系统或环境的变化。这是因为自适应滤波器的参数是在不断变化以满足某种性能要求,所以它是时变滤波器。

在自适应滤波理论体系中,最小均方(LMS, least-mean square)算法是一种搜索算法,它通过对目标函数进行适当的调整,简化了对梯度向量的计算。因 LMS 算法具有低计算复杂度、在平稳环境中的收敛性好、其均值无偏地收敛到 Wiener 解和利用有限精度实现算法时的稳定性等特性,使 LMS 算法成为自适应算法中应用最广泛的算法。

## 1.2 自适应 LMS 算法的发展

### 1.2.1 LMS 算法的历史

1955-1966 年期间,美国通用公司在研制天线的过程中,为抑制旁瓣,由 Windrow 和 Hoff 在 60 年代初提出了基本 LMS 算法。随后又发展出了归一化 LMS 算法和加遗忘因子 LMS 算法。1977 年, Makhoul 提出了格型滤波器,并由此发展出 LMS 自适应格型滤波器算法。Herzberg、Cohen 和 Be'ery 提出了延时 LMS (DLMS) 算法。2002 年,尚勇,吴顺君,项海格提出了并行延时 LMS 算法。此

外, 还有复数 LMS 算法、数据块 LMS 算法等, 在此就不一一列举了。

### 1.2.2 LMS 算法的现状

因 LMS 算法具有低计算复杂度、在平稳环境中的收敛性好、其均值无偏地收敛到 Wiener 解和利用有限精度实现算法时的稳定性等特性, 使 LMS 算法成为自适应算法中应用最广泛的算法。又由于 LMS 算法的广泛应用, 以及在实际条件下, 为解决实际问题, 基于 LMS 算法的新 LMS 类算法不断出现。

### 1.2.3 LMS 算法的发展前景

因 LMS 算法是自适应滤波器中应用最广泛的算法, 所以可以说, 自适应滤波的发展前景也就是 LMS 算法的发展前景。它主要包括以下几个方面的应用:

1、系统辨识和建模(System Identification and Modeling)。自适应滤波器作为估计未知系统特性的模型。

2、自适应信道均衡(Adaptive Channel Equalization)。在数字通信中采用自适应信道均衡器, 可以减小传输失真, 以及尽可能地利用信道带宽。

3、回波消除(Echo Cancellation)。在 2 线和 4 线环路电话系统中, 线路间存在杂散电路耦合, 这些杂散导致阻抗不匹配, 从而形成了信号的反射, 也就是我们在线路两端听到的回声。这种回波能对高速数据传输造成灾难性的后果。回波消除就是预先估计一个回波, 然后用返回信号来减此回波, 从而达到回波消除的目的。消除心电图中的电源干扰就是它的一个具体应用。

4、线性预测编码(Linear Predictive Coding)。近年来, 对语音波形进行数字编码成为流行, 通过分析语音波形来获得语音的基本参数, 然后对语音波形进行编码, 它可以大大降低数据传输率。在接收端使用 LPC 分析得到的参数, 通过语音合成器重构语音。合成器实际上是一个离散的随时间变化的时变线性滤波器。时变线性滤波器既当作预测器使用, 又当作合成器使用。分析语音波形时作预测器使用, 合成语音时作语音生成模型使用。

5、自适应波束形成(Adaptive Beamforming)。频谱资源越来越紧张, 利用现有频谱资源进一步扩展容量成为通信发展的一个重要问题。智能天线技术利用阵列天线替代常规天线, 它能够降低系统干扰, 提高系统容量和频谱效率,

因此智能天线技术受到广泛关注。自适应波束形成通过调节天线各阵元的加权幅度和相位，来改变阵列的方向图，使阵列天线的主瓣对准期望用户，从而提高接收信噪比，满足某一准则下的最佳接收。在雷达与声纳的波束形成中，自适应滤波器用于波束方向控制，并可在波束方向图中提供一个零点以便消除不希望的干扰。

其应用还有噪声中信号的滤波、跟踪、谱线增强以及线性预测等。

### 1.3 本论文的主要研究内容和创新之处

#### 1.3.1 本论文的主要研究内容

本文的主要是研究 LMS 算法，包括基本 LMS 算法、归一化 LMS 算法、加遗忘因子 LMS 算法、解相关 LMS 算法、并行延时 LMS 算法、自适应格型 LMS 算法、Newton-LMS 算法和，并对各种 LMS 算法进行了性能分析、对比。此外，还给出了 LMS 算法在励磁电源的自动控制系统中的实现方法。

#### 1.3.2 本论文的创新之处

本文在设计研究中做了以下工作和创新：

1、对 LMS 算法的历史和发展前景进行了综述。

2、分别介绍了几种主要的 LMS 算法，并通过 Matlab 实验仿真，给出了仿真图形，分析了它们的性能。

3、对 LMS 算法的参数进行了分析，并通过 Matlab 仿真，用仿真结果来说明各种参数在 LMS 算法中的作用。

4、分析了几种对 LMS 算法有影响的因素，并把其对 LMS 算法的影响通过仿真直观地展现出来。

5、结合实际，把 LMS 算法应用到自动控制系统中。在本文的励磁电源控制项目中，由于现场环境恶劣且变化很大，用单片机很难满足要求，我们采用 DSP 并结合自适应 LMS 算法对干扰进行抑制，取得了很好的效果。

## 2 自适应 LMS 算法的研究

### 2.1 概述

自适应算法中使用最广的是下降算法，下降算法的实现方式有两种：自适应梯度算法和自适应高斯-牛顿算法。自适应高斯-牛顿算法包括 RLS 算法及其

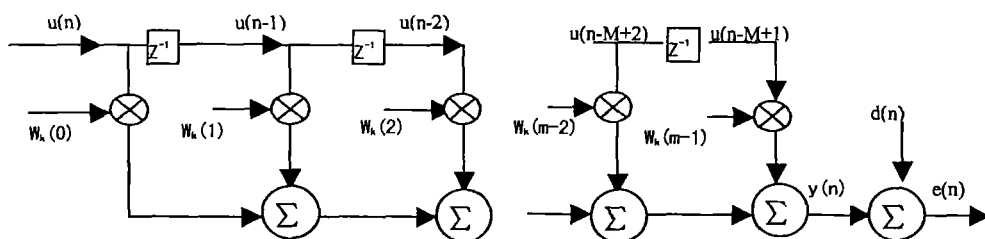


图 2.1 FIR 滤波器的自适应实现

变型和改进型，自适应梯度算法包括 LMS 算法及其变型和改进型。

滤波器设计准则是使滤波器实际输出  $y(n)$  与期望响应  $d(n)$  之间的均方误差  $J(n)$  为最小，这称为最小均方误差（MMSE）准则。

图 2.1 为 FIR 滤波器的自适应实现的原理图。所谓自适应实现是指：M 阶 FIR 滤波器的抽头权系数  $w_0, w_1, \dots, w_{m-1}$  可以根据估计误差  $e(n)$  的大小自动调节，使得某个代价函数最小。

定义均方误差  $J(n)$  为代价函数，因为滤波器在  $n$  时刻的估计误差

$$e(n) = d(n) - w^H u(n) \quad (2.1)$$

所以代价函数

$$J(n) = E\{|e(n)|^2\} = E\{|d(n) - w^H u(n)|^2\} \quad (2.2)$$

由此可得  $J(n)$  的梯度（具体推导见参考文献[1]）

$$\nabla J(n) = 2 E\{u(n) u^H(n)\} w(n) - 2 E\{u(n) d^*(n)\} \quad (2.3)$$

### 2.2 LMS 算法及其基本变型

自适应算法中最常用的下降算法为最陡下降算法，其更新方向向量取第  $n$

次迭代的代价函数  $J(n)$  的负梯度，它的统一形式为

$$w(n+1) = w(n) - \mu \nabla J(n) \quad (2.4)$$

式中  $w(n)$  为第  $n$  次迭代的权向量， $-\nabla J(n)$  为第  $n$  次迭代的更新方向向量， $\mu$  为第  $n$  次迭代的更新步长，它用来控制稳定性和收敛速度。

若将  $\nabla J(n)$  中的真实梯度向量用瞬时梯度向量代替，我们可得到由 Windrow 和 Hoff 在 60 年代初提出的著名的最小均方误差自适应算法（简称 LMS 算法）：

$$w(n+1) = w(n) + \mu [d(n+1) - w^H(n)u(n+1)]^* u(n+1) \quad (2.5)$$

上式表明，LMS 算法是一种递归运算，它不需要对信号的统计特性有先验的了解，而只是使用它们的瞬时估计值，运算得到的只是权重系数的估计值，但随着时间的增加，权重系数逐步调整，估计值也逐步改善，最终得到收敛值。收敛的条件是：

$$0 < \mu < \frac{1}{\lambda_{\max}} \quad (2.6)$$

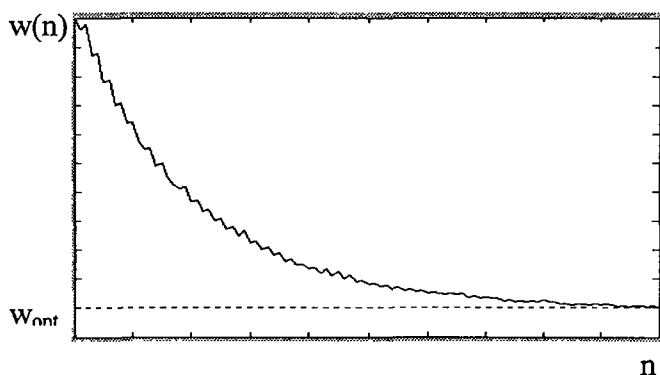


图 2.2  $w(n)$  自适应调整过程

式中  $\lambda_{\max}$  是输入数据协方差矩阵的本征值。实际上， $w(n)$  永远不会收敛到理论上的最佳值，而是在最佳值上下振荡，如图 2.2 所示。 $\mu$  值控制了自适应的“增益”，它的大小应该在算法的收敛性能和系数的振荡幅度这两个因素之间折中。

LMS 算法简单而且易于实现，，是许多实时应用的首选算法，它对每一次新

的输入和输出采样需要作大约  $2N+1$  次乘-加运算。

### 2.2.1 基本 LMS 算法

其运算步骤如下：

①初始化设置  $w(n)$  ( $n=0, 1, 2, \dots, N-1$ ) 为任意值（一般均为零），然后对每一次采样作以下各步的循环运算：

②计算滤波器输出：
$$y(n) = \sum_{i=0}^{N-1} w(n)u(n-i)$$

③计算估计误差：
$$e(n) = d(n) - y(n)$$

④更新  $N$  个滤波器权重系数：

$$w(n) = w(n-1) + \mu u(n) e^*(n) \quad (n=0, 1, 2, \dots, N-1)$$

⑤循环返回到②。

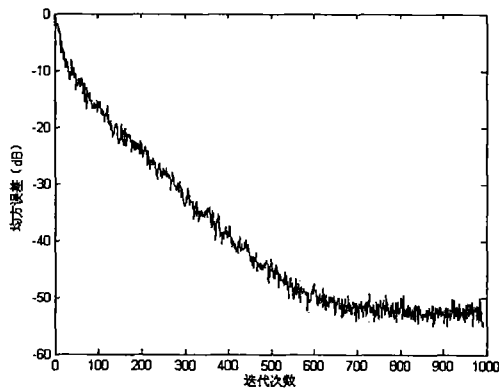


图 2.3 基本 LMS 算法的学习曲线

式中  $w(n)$  为抽头权向量， $e(n)$  为滤波器在  $n$  时刻的估计误差， $\mu$  为常数， $d(n)$  为期望响应，当期望信号未知时，可直接用滤波器的实际输出  $y(n)$  代替  $d(n)$ ， $u(n)$  为输入信号。

当  $\mu=0.01$ 、滤波器阶数为 32 阶，输入信号为符号函数

$$\text{sign}(x) = \begin{cases} 1, x > 0 \\ 0, x = 0 \\ -1, x < 0 \end{cases} \quad (2.7)$$

时，得到基本 LMS 算法的学习曲线如图 2.3。

### 2.2.2 归一化 LMS 算法

$$w(n) = w(n-1) + \frac{\alpha}{\beta + u^H(n)u(n)} u(n) e^*(n) \quad n=1, 2, 3, \dots \quad (2.8)$$

$$e(n) = d(n) - w^H(n-1)u(n) \quad (2.9)$$

$\alpha$ ， $\beta$  为常数且  $\alpha \in (0, 2)$ ， $\beta \geq 0$ 。

其运算步骤同基本 LMS 算法。图 2.4 是在  $\alpha=0.3$ ， $\beta=1$  时得到的学习速率

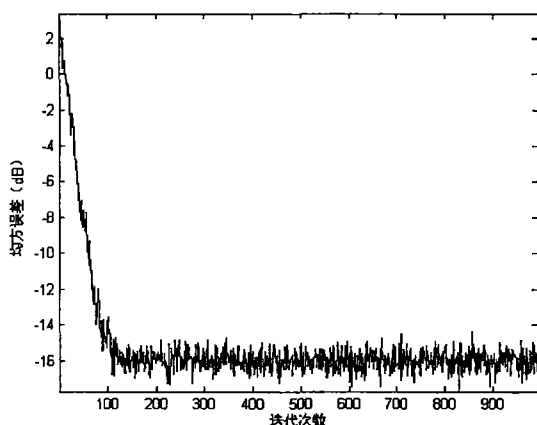


图 2.4 归一化 LMS 算法的学习曲线

曲线，对比图 2.3 与图 2.4，可以明显看出归一化 LMS 算法的收敛速度比基本 LMS 算法的收敛速度快得多。

### 2.2.3 加遗忘因子 LMS 算法

$$w(n) = w(n-1) + \frac{\alpha}{\beta(n)} u(n) e^*(n) \quad n=1, 2, 3, \dots \quad (2.10)$$

$$e(n) = d(n) - w^H(n-1)u(n) \quad (2.11)$$

$\alpha$  为常数且  $\alpha \in (0, 2)$ ,  $\beta(n)$  表示  $u(n)$  的方差, 可由下式递推计算:  
 $\beta(n) = \lambda \beta(n-1) + e^2(n)$ ,  $\lambda$  为遗忘因子,  $\lambda \in (0, 1)$ 。

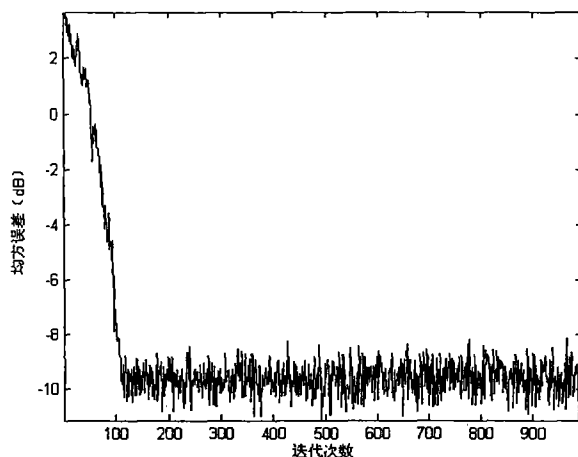


图 2.5 加速遗忘因子 LMS 算法的学习曲线

其运算步骤同基本 LMS 算法。图 2.5 是在  $\alpha=0.03$ ,  $\lambda=0.5$  的条件下的学习速率曲线。从图中可以看出加速遗忘因子 LMS 算法收敛速度也比基本 LMS 算法快。

### 2.3 量化误差 LMS 算法

在回声消除和信道均衡等需要自适应滤波器高速工作的应用中, 降低计算复杂度是很重要的。LMS 算法的计算复杂度主要来自在进行数据更新时的乘法运算以及对自适应滤波器输出的计算, 量化误差 LMS 算法就是一种降低计算复杂度的方法。其基本思想是对误差信号进行量化, 其滤波器系数更新公式如下:

$$w(n+1) = w(n) + 2\mu Q[e(n)]x(n) \quad (2.12)$$

式中函数  $Q[*]$  代表量化运算, 不同的量化函数  $Q[*]$  确定了不同的量化误差 LMS 算法。

常见的量化误差 LMS 算法有符号误差 LMS 算法、符号-数据 LMS 算法、双符号 LMS 算法、2 的幂误差 LMS 算法等, 本文讨论符号误差 LMS 算法和符号数据



LMS 算法。

### 2.3.1 符号误差 LMS 算法

$$\text{当量化函数 } Q[*] \text{ 采用符号函数 } \text{sign}(x) = \begin{cases} 1, x > 0 \\ 0, x = 0 \\ -1, x < 0 \end{cases}$$

时，量化误差 LMS 算法称为符号误差 LMS 算法。其系数向量的更新方程为

$$w(n+1) = w(n) + 2\mu \text{sign}[e(n)] x(n) \quad (2.13)$$

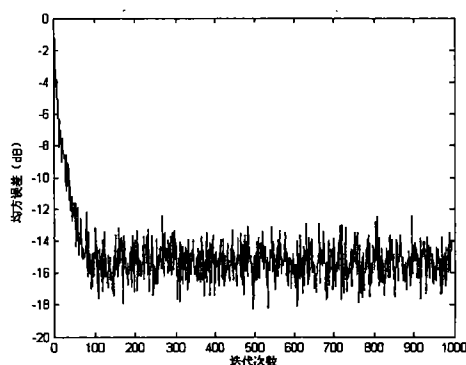


图 2.6 符号误差 LMS 算法的学习曲线

图 2.6 为符号误差 LMS 算法的学习曲线，大约迭代 100 次就达到收敛。收敛速度明显比基本 LMS 算法快。

### 2.3.2 符号数据 LMS 算法

为减轻 LMS 算法的计算复杂度，除对误差  $e(n)$  进行量化这种方法外，还有对输入数据  $x(n)$  进行量化。从本质上说，符号数据算法不是量化误差算法。其系数更新公式为

$$w(n+1) = w(n) + 2\mu e(n) \text{sign}[x(n)] \quad (2.14)$$

对数据向量的量化可能使收敛速率降低，并且还可能导致发散。在符号-数据算法中，平均梯度方向只能沿着某些离散的方向进行更新，这可能引起更新过程中平方误差的频繁增加，从而导致不稳定。

## 2.4 解相关 LMS 算法

在 LMS 算法中，有一个独立性假设：横向滤波器的输入  $u(1), u(2), \dots, u(n-1), u(n)$  是彼此统计独立的向量序列。当它们之间不满足统计独立的条件时，基本 LMS 算法的性能将下降，尤其是收敛速度会比较慢。为解决此问题，提出了解相关 LMS 算法。研究表明（见参考文献[2]），解相关能够有效加快 LMS 算法的收敛速度。解相关 LMS 算法又分为时域解相关 LMS 算法和变换域解相关 LMS 算法，下面分别进行讨论。

### 2.4.1 时域解相关 LMS 算法原理

我们定义  $u(n)$  与  $u(n-1)$  在  $n$  时刻的相关系数为

$$\alpha(n) = \frac{u^H(n-1)u(n)}{u^H(n-1)u(n-1)} \quad (2.15)$$

若  $\alpha(n)=1$ ，则称  $u(n)$  是  $u(n-1)$  的相干信号；若  $\alpha(n)=0$ ，则称  $u(n)$  与  $u(n-1)$  不相关；当  $0 < \alpha(n) < 1$  时，称  $u(n)$  与  $u(n-1)$  相关，并且  $\alpha(n)$  越大， $u(n)$  与  $u(n-1)$  之间的相关性越强。

显然， $\alpha(n)u(n-1)$  代表了  $u(n)$  与  $u(n-1)$  相关的部分，若从  $u(n)$  中减去该部分，则这一减法运算相当于“解相关”，用解相关的结果作为更新方向向量：

$$v(n) = u(n) - \alpha(n)u(n-1) \quad (2.16)$$

从上式可以看出， $\alpha(n)$  为解相关系数。

另一方面，收敛因子  $\mu(n)$  应该是满足下式最小化问题的解

$$\mu(n) = \arg \min_{\mu} J[w(n-1) + \mu v(n)] \quad (2.17)$$

由此可得

$$\mu(n) = \frac{e(n)}{u^H(n)v(n)} \quad (2.18)$$

由上面的结果，我们可以得到解相关 LMS 算法，它是 Doherty 与 Porayath 于 1997 年提出的。其算法如下：

①初始化： $w(0)=0$ ；

②更新:  $n=1, 2, \dots$

$$\begin{aligned} e(n) &= d(n) - w^H(n-1)u(n) \\ \alpha(n) &= \frac{u^H(n-1)u(n)}{u^H(n-1)u(n-1)} \\ v(n) &= u(n) - \alpha(n)u(n-1) \\ \mu(n) &= \frac{\rho e(n)}{u^H(n)v(n)} \\ w(n) &= w(n-1) + \mu(n)v(n) \end{aligned}$$

式中参数  $\rho$  称为修整因子。

#### 2.4.2 变换域解相关 LMS 算法

此算法的基本思想是对输入信号进行变换, 使得输入信号变为不相关。这是为什么把变换域 LMS 算法也看作是一种解相关算法的原因。为去除输入信号的相关性, 可以通过采用正交变换或者酉变换, 使输入信号向量  $x(n)$  变换为另一个向量  $s(n)$

$$s(n) = T x(n) \quad (2.19)$$

式中  $TT^T = I$ 。下图给出了变换域自适应滤波器的结构。

文献[11]对变换域 LMS 算法原理有详细的推导, 这里就不再重复, 直接给出完整的变换域 LMS 算法

初始化:  $x(0) = w(0) = [0 \dots 0]^T$

$\gamma$  = 小常数,  $0 < \alpha \leq 0.1$

$$s(n) = T x(n)$$

$$e(n) = d(n) - s^T(n)w(n)$$

$$w(n+1) = w(n) + 2\mu e(n)\sigma^{-2}(n)s(n)$$

式中  $\sigma^{-2}(n)$  是一对角矩阵, 其元素值是  $s(n)$  的元素的功率估计值加上  $\gamma$  以后的倒

数。

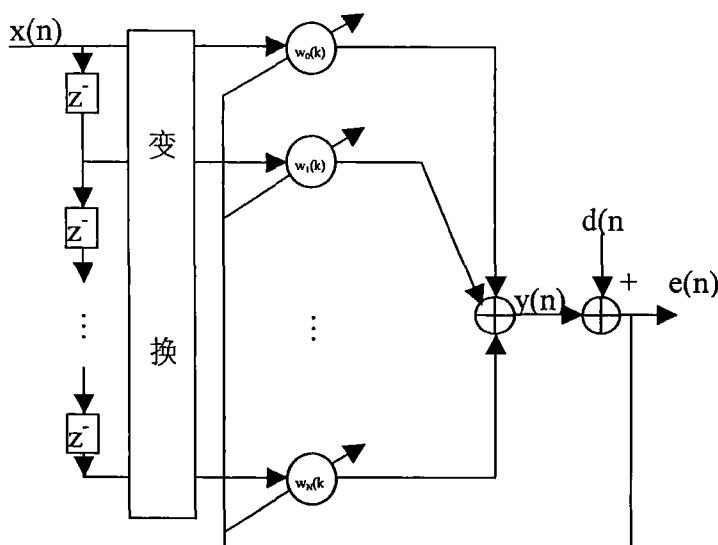


图 2.7 变换域滤波器结构图

## 2.5 并行延时 LMS 算法

在自适应算法的实现结构中，有一类面向 VLSI 的脉动（systolic）结构，由于其具有的高度并行性和流水线特性而备受关注。将 LMS 算法直接映射到脉动结构时，在权值更新和误差计算中存在着严重的计算瓶颈，为了克服这种瓶颈效应，Herzberg、Cohen 和 Be'ery 提出了延时 LMS (DLMS) 算法(详见参考文献[8])，Long、Ling 和 Proakis 详细分析了延时 LMS 算法的性能(详见参考文献[6]、[7])。该算法解决了算法到结构的计算瓶颈问题，但当滤波器阶数较长时，算法的收敛性能会变差，这是由于其本身所具有的延时影响了它的收敛性能。可以说，延时 LMS 算法是以牺牲算法的收敛性能为代价的。

针对延时 LMS 算法在收敛性能方面的不足，人们提出了一些改进算法(见参考文献[9]、[10])，这些算法在一定程度上改善了延时 LMS 算法的收敛性，但同时在算法实现结构上增加了复杂度。尚勇，吴顺君，项海格提出了一种将延时技术和并行技术相结合的自适应 LMS 算法——PDLMS 算法。与延时 LMS 算法相比，在相同条件下，并行延时 LMS 算法保证算法收敛的步长因子上限更大，算

法收敛速度更快，数据吞吐率更高，对相关输入有一定的去相关作用，同时保留了延时 LMS 算法在实现结构上简单、高效的优点。特别适合于滤波器阶数高，要求系统收敛速度快的应用场合，还可以直接映射到具有高速、高流水特性的硬件结构中。

### 2.5.1 延时 LMS 算法

对延时 LMS 算法而言，最佳步长  $\mu$  的选择与输入信号的延时量大小有关，其算法原理如下：

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \mu e(n-m) \mathbf{x}(n-m) \quad (2.20)$$

$$e(n) = d(n) - y(n) \quad (2.21)$$

$$y(n) = \mathbf{x}^H(n) \mathbf{w}(n-1) \quad (2.22)$$

其中

$$\mathbf{w}(n) = \begin{bmatrix} w_{0,n} \\ w_{1,n} \\ \vdots \\ w_{N-1,n} \end{bmatrix}, \mathbf{x}(n) = \begin{bmatrix} x(n) \\ x(n-1) \\ \vdots \\ x(n-N+1) \end{bmatrix}$$

式中， $n$  代表采样时刻， $m$  为延时量， $N$  为滤波器长度， $d(n)$ 、 $y(n)$  和  $e(n)$  分别为  $n$  时刻的期望信号、输出信号和误差信号。由式  $\mathbf{w}(n) = \mathbf{w}(n-1) + \mu e(n-m) \mathbf{x}(n-m)$  可知， $n$  时刻权值  $\mathbf{w}(n)$  的更新采用的是延时  $m$  时刻的误差信号  $e(n-m)$  和  $\mathbf{x}(n-m)$ 。

延时 LMS 算法性能的详细分析见文献[6]和[7]。文献[6]和[7]给出了不同近似条件下，保证算法收敛的步长因子的选择范围和相应的失调量。从文献的结果可以看出，在同一情况下，无论采用何种近似，随着  $m$  的增加， $\mu$  限逐渐减小。由于 LMS 算法可以看做当  $m=0$  时的特殊情况，因此，在相同条件下，LMS 算法的  $\mu$  限比延时 LMS 算法大。

### 2.5.2 并行延时 LMS 算法

对于二输入二输出并行滤波器结构，对应每一条输出支路都有两条输入支路用作自适应处理，延时技术在其中也分为两部分处理。

设原型滤波器长度为  $N$ ，此时

$$\begin{bmatrix} w_{0,n} \\ w_{1,n} \end{bmatrix} = \begin{bmatrix} w_{0,n-1} \\ w_{1,n-1} \end{bmatrix} + 2\mu e_{1,n-m} \begin{bmatrix} X_{1,n-m} \\ X_{0,n-m} \end{bmatrix} \quad (2.23)$$

$$y_{1,n} = \mathbf{w}_{0,n}^H \mathbf{X}_{1,n} + \mathbf{w}_{1,n}^H \mathbf{X}_{0,n} \quad (2.24)$$

$$e_{1,n} = d_{1,n} - y_{1,n} = d_{1,n} - \mathbf{h}_0^H \mathbf{X}_{1,n} - \mathbf{w}_{1,n}^H \mathbf{X}_{0,n} \quad (2.25)$$

$$\text{令 } \mathbf{R}_{11} = E\{\mathbf{X}_{1,n} \mathbf{X}_{1,n}^H\}, \quad \mathbf{R}_{00} = E\{\mathbf{X}_{0,n} \mathbf{X}_{0,n}^H\}, \quad \mathbf{P}_{10} = E\{d_{1,n} \mathbf{X}_{0,n}\}, \quad \mathbf{P}_{11} = E\{X_{1,n} X_{1,n}\},$$

则  $\mathbf{h}_0$  和  $\mathbf{h}_1$  的最优权值分别为

$$\mathbf{h}_0^* = \mathbf{R}_{11}^{-1} \mathbf{P}_{11}, \quad \mathbf{h}_1^* = \mathbf{R}_0^{-1} \mathbf{P}_{10} \quad (2.26)$$

此时

$$\begin{aligned} e_{1,n-m} &= d_{1,n-m} - \mathbf{h}_{0,n-m-1}^H \mathbf{X}_{1,n-m} - \mathbf{h}_{1,n-m-1}^H \mathbf{X}_{0,n-m} \\ &= e_{1,n-m}^* - \mathbf{V}_{0,n-m-1}^H \mathbf{X}_{1,n-m} - \mathbf{V}_{1,n-m-1}^H \mathbf{X}_{0,n-m} \end{aligned} \quad (2.27)$$

其中

$$\mathbf{V}_{0,n-m-1}^H = \mathbf{h}_{0,n-m-1}^H \mathbf{h}_0^*$$

$$\mathbf{V}_{1,n-m-1}^H = \mathbf{h}_{1,n-m-1}^H \mathbf{h}_1^*$$

$e_{1,n-m}^*$  为最优权对应的误差信号。

### 2.5.3 并行延时 LMS 算法性能分析

下面以  $y_{1,n}$  支路为例，对并行延时 LMS 算法的性能进行分析。这里重点讨论保证算法收敛的步长因子  $\mu$  的范围和系统的失调量  $M$ 。假设输入数据与权向量以及误差信号互不相关，且信号是均值为零的高斯过程。

当系统达到稳态时  $\xi$  的表达式（详细推导见参考文献[5]）

$$\xi = \frac{\alpha_1 N b_1 E\{e_{1,n}^2\}_{\min}}{2 - (\alpha_1 K_1 + 2m + \frac{N}{2} \alpha_1) b_1 + (m^2 + m) b_1^2 + \dots} \quad (2.28)$$

对应的失调量

$$M = \frac{\alpha_1 N b_1}{2 - (\alpha_1 K_1 + 2m + \frac{N}{2} \alpha_1) b_1 + (m^2 + m) b_1^2 + \dots} \quad (2.29)$$

式中

$$\alpha_1 = \frac{\lambda_{1,rms}^2}{\lambda_{1,av}^2}, \lambda_{1,rms}^2 = \frac{\sum_{i=1}^{N/2} \lambda_{1,i}^2}{N/2}, \lambda_{1,av} = \sigma^2 = \frac{\sum_{i=1}^{N/2} \lambda_{1,i}}{N/2}. \quad (2.30)$$

根据式 (2.28)，可以得到在不同近似条件下的并行延时 LMS 算法的步长因子  $\mu$  的上限。 $\mu$  由下面三式决定：

$$\begin{aligned} 0 < \mu &< \frac{2}{(\alpha_1 K_1 + 2m + \frac{N}{2} \alpha_1) \sigma_1^2} \\ 0 < \mu &< \frac{\alpha_1 K_1 + 2m + \frac{N}{2} \alpha_1 - \sqrt{(\alpha_1 K_1 + 2m + \frac{N}{2} \alpha_1)^2 - 8m(m+1)}}{2m(m+1)\sigma_1^2} \\ 0 < \frac{2}{\sigma_1^2} - (\alpha_1 K_1 + 2m + \frac{N}{2} \alpha_1) \mu + m(m+1) \mu^2 - \frac{1}{3} m(m+1)(2m+1) \mu^3 \end{aligned}$$

上面各式中的参数按式 (2.30) 选取。

上面是以  $y_{1,n}$  支路为例对并行延时 LMS 算法性能进行的分析，若对  $y_{0,n}$  支路进行分析可得到相似的结论。

图 2.8 与图 2.9 是在输入一组不相关数据的条件下，其中图 2.8 中步长因子  $\mu=0.01$ ，延时 LMS 算法的延时量  $D=18$ ；并行延时 LMS 算法的延时量  $D=9$ ，各子滤波器长度为 9。图 2.9 中步长因子  $\mu=0.1$ ，延时 LMS 算法的延时量  $D=12$ ；并行延时 LMS 算法的延时量  $D=5$ ，各子滤波器长度为 9。

从图 2.8 与图 2.9 可以看出，当输入数据为不相关的数据时，在算法收敛的情况下，并行延时 LMS 算法的收敛速度比延时 LMS 算法的快。当步长因子  $\mu$  增大时，并行延时 LMS 算法仍能收敛，而延时 LMS 算法已不能很好地收敛。

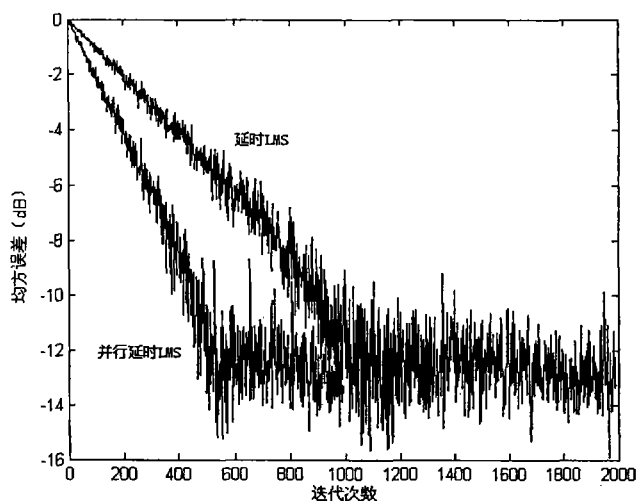


图 2.8 并行 LMS 与延时 LMS 算法收敛速度比较

并行延时 LMS 算法具有良好收敛特性的原因，一个是因为其子滤波器长度只有原型滤波器的一半，因而所引入的延时量也只有延时 LMS 算法的一半；另一个是由于并行延时 LMS 算法的权值更新速率只为采样率的一半，这对相关输入信号有一定的去相关作用，这种去相关作用有助于进一步改善并行延时 LMS 算法的收敛性。

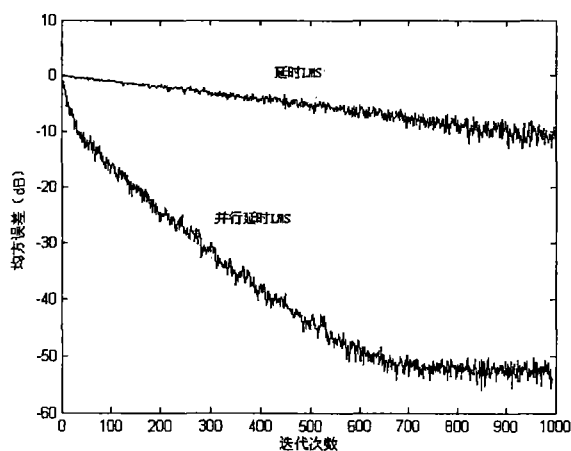


图 2.9 并行 LMS 与延时 LMS 算法收敛速度比较



## 2.6 自适应格型 LMS 算法

LMS 滤波器属于横向自适应滤波器且假定阶数固定，然而在实际应用中，横向滤波器的最优阶数往往是未知的，需要通过比较不同阶数的滤波器来确定最优的阶数。当改变横向滤波器的阶数时，LMS 算法必须重新运行，这显然不方便而且费时。那么，在增加滤波器的阶数时，能否利用低一阶滤波器参数的结果呢？格型滤波器解决了这一问题。

格型滤波器最早是 Makhoul 于 1977 年提出的，当时将 Makhoul 采用的方法称为线性预测的格型方法，后来才称为格型滤波器。格型滤波器具有共轭对称的结构，前向反射系数是后向反射系数的共轭，其设计准则和 LMS 算法一样是使均方误差最小。

### 2.6.1 对称的格型结构

LMS 自适应格型滤波器的结构是在每一级对前、后向分别采用反射系数  $r_p^*$  和  $r_p$ ，如图 2.3 所示。图中  $f_p(n)$  和  $g_p(n)$  分别是第  $p$  级格型滤波器的前向残差和后向残差。

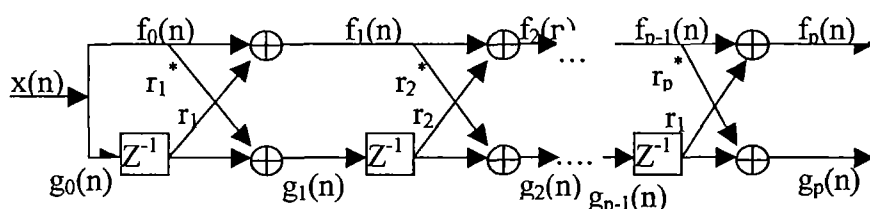


图 2.10 LMS 自适应格型滤波器

由图可知， $n$  时刻的前向残差和后向残差服从以下递推关系

$$f_m(n) = f_{m-1}(n) + r_m g_{m-1}(n-1) \quad (2.31)$$

$$g_m(n) = r_m^* f_{m-1}(n) + g_{m-1}(n-1) \quad (2.32)$$

其初始值为

$$f_0(n) = g_0(n) = x(n)$$

由于  $f_m(n)$  和  $g_m(n)$  通过  $r_m$  建立了关系, 所以也把  $r_m$  称为偏相关系数。

此外, 后向滤波器权系数  $b_m$  与前向滤波器权系数  $a_m$  存在以下关系 (详细推导见文献[1])

$$b_m(i) = a_m^*(m-i), \quad i=0, 1, \dots, m$$

上式说明, 只要前向滤波器系数  $a_m(i)$  已经设计出来, 即可确定后向滤波器的系数  $b_m(i)$ 。于是格型滤波器的设计便归结为前向滤波器设计。

## 2.6.2 格型滤波器设计准则

因为格型滤波器的设计可归结为前向滤波器设计, 这里就只讨论前向滤波器设计准则。

格型滤波器  $n$  时刻的前向残差的  $Z$  变换

$$F_m(z) = F_{m-1}(z) + r_m z^{-1} G_{m-1}(z) \quad (2.33)$$

可等价于

$$F_m(z) = A_m(z) X(z) \quad (2.34)$$

相应的时域表示式为

$$f_m(n) = \sum_{i=0}^m a_m(i) x(n-i) \quad (2.35)$$

前向滤波器的残差能量

$$\begin{aligned} F_m &= E\{|f_m(n)|^2\} \\ &= \sum_{i=0}^m \sum_{j=0}^m a_m(i) a_m^*(j) E\{x(n-i) x^*(n-j)\} \\ &= \sum_{i=0}^m \sum_{j=0}^m a_m(i) a_m^*(j) R_x(j-i) \end{aligned} \quad (2.36)$$

式中  $R_x(j-i) = E\{x(n)x^*(n-j+i)\}$  是滤波器输入信号  $x(n)$  的相关函数。

为确定前向滤波器  $A(i)$  的系数, 只需使残差能量  $F$  最小即可。因为

$$\frac{\partial F_m}{\partial a_m^*(j)} = 0, \quad j=1, 2, \dots, m \quad (2.37)$$

所以由式(2.36)和(2.37)得

$$\sum_{i=0}^m a_m(i)R_x(j-i)=0, \quad j=1, 2, \dots, m \quad (2.38)$$

将上式代入式(2.36)得

$$\begin{aligned} F_m &= \sum_{i=0}^m a_m(i)R_x(-i) + \sum_{j=1}^m a_m^*(j) \sum_{i=0}^m a_m(i)R_x(j-i) \\ &= \sum_{i=0}^m a_m(i)R_x(-i) \end{aligned} \quad (2.39)$$

理论上, 求解式(2.37)可得到  $m$  级前向滤波器的系数  $a_m(1), a_m(2), \dots, a_m(m)$ , 然后将这些系数代入式(2.37)就可得到前向残差能量。一般情况下, 阶数  $m$  越大, 前向残差能量  $F_m$  越小。

在设计前向滤波器时, 当前向残差能量  $F_m$  不再明显减小时, 最小的阶数  $m$  即为格型滤波器的最优阶数。但有一个问题: 格型滤波器前后级间是否存在着耦合? 文献[1]P218-P221 证明了不同级滤波器的后向残差正交, 这说明格型滤波器前后级是解耦的, 从而可以独立地设计每一级滤波器。

### 2.6.3 自适应格型 LMS 算法

令  $w(n)$  为滤波器在  $n$  时刻的权系数, 并且  $w(n)$  满足

$$\begin{cases} w(n)=0, & n < 0 \\ w(n) > 0, & n \geq 0 \end{cases} \quad (2.40)$$

定义瞬时前向、后向残差能量

$$e_m(k) = (1 - \beta) |f_m(k)|^2 + \beta |g_m(k)|^2, 0 \leq \beta \leq 1$$

和  $n$  时刻及以前所有时刻前向、后向残差的加权总能量误差函数

$$E_m(n) = \sum_{k=-\infty}^n w(n-k) e_m(k) \quad (2.41)$$

最优反射系数  $r_m$  可利用下式确定

$$\begin{aligned} \frac{\partial E_m(n)}{\partial r_m^*} &= \sum_{k=-\infty}^{\infty} w(n-k) \frac{\partial}{\partial r_m^*} [(1 - \beta) |f_m(k)|^2 + \beta |g_m(k)|^2] \\ &= \sum_{k=-\infty}^{\infty} w(n-k) [(1 - \beta) f_m(k) g_{m-1}^*(k-1) + \beta g_m^*(k) f_{m-1}(k)] \\ &= 0 \end{aligned}$$

整理后得

$$r_m(n) = \frac{- \sum_{k=-\infty}^{\infty} w(n-k) f_{m-1}(k) g_{m-1}^*(k-1)}{\sum_{k=-\infty}^{\infty} w(n-k) [\beta |f_m(k)|^2 + (1 + \beta) |g_m(k-1)|^2]} \quad (2.42)$$

且

$$|r_m(n)| < 1$$

这一条件保证了  $m+1$  阶前向滤波器第  $m+1$  个系数在任意时刻  $n$  的值都能够满足  $|a_{m+1}(m+1)| < 1$  的条件, 从而使得前向滤波器是最小相位的(即物理可实现的)。

与 Burg 最大熵方法中滤波器系数的估计式

$$k_m(n) = \frac{- \sum_{n=m+1}^N f_{m-1}(n) g_{m-1}^*(n-1)}{\frac{1}{2} \sum_{n=m+1}^N [|f_{m-1}(n)|^2 + |g_{m-1}(n-1)|^2]} \quad (2.43)$$

相比较知, 若  $\beta=0.5$  及  $w(k)\equiv 1$ , 则式(2.41)简化为 Burg 最大熵方法中滤波器前向、后向预测误差总的平均能量, 滤波器系数的最大熵估计由式(2.42)给出。

若引入符号

$$C_{m-1}(n) = \sum_{k=-\infty}^{\infty} w(n-k) f_{m-1}(k) g_{m-1}^*(k-1)$$

$$D_{m-1}(n) = \sum_{k=-\infty}^{\infty} w(n-k) [\beta |f_{m-1}(k)|^2 + (1-\beta) |g_{m-1}(k-1)|^2]$$

则

$$r_m(n) = -\frac{C_{m-1}(n)}{D_{m-1}(n)} \quad (2.44)$$

并且  $C_{m-1}(n)$  和  $D_{m-1}(n)$  服从以下递推公式

$$C_{m-1}(n) = D_{m-1}(n-1) + w(0) f_{m-1}(n) g_{m-1}^*(n-1) \quad (2.45)$$

$$D_{m-1}(n) = D_{m-1}(n-1) + w(0) [\beta |f_{m-1}(k)|^2 + (1-\beta) |g_{m-1}(k-1)|^2] \quad (2.46)$$

由式(2.31)、(2.32)、(2.44)、(2.45)和(2.46)以及下面三式可组成完整的 LMS 格型自适应滤波算法

$$a_m^{(m)}(n) = r_m(n)$$

$$a_i^{(m)}(n) = a_i^{(m-1)}(n) + r_m(n) a_{m-i}^{*(m-1)}(n)$$

$$P_m = (1 - |r_m|^2) P_{m-1}$$

式中  $a_i^{(m)}(n)$  表示  $m$  阶前向滤波器第  $i$  个系数在  $n$  时刻的值,  $P_m$  为  $m$  阶格型滤波器的残差能量。

上面用递推公式获得滤波器系数的同时, 还可得到各阶滤波器的残差能量, 当残差能量不再减小时的最小阶数即为 LMS 格型滤波器的最优阶数。

由上面的分析可以看出, LMS 格型自适应滤波器的优点是收敛速度比 LMS 横向滤波器快得多, 而且对数据的舍入误差不敏感, 其代价是需要比 LMS 算法更大的计算量。

## 2.7 Newton-LMS 算法

本算法是对环境信号二阶统计量进行估计的算法。其目的是为了解决输入信号相关性很高时 LMS 算法收敛速度慢的问题。一般情况下, 牛顿算法能够快速收敛, 但对  $R^{-1}$  的估计所需计算量很大, 而且存在数值不稳定的问题, 这导致了基于最陡下降的算法在自适应滤波中应用更广泛。

### 2.7.1 算法原理

如果牛顿更新公式

$$w(n+1) = w(n) - \mu R^{-1} g_w(n) \quad (2.47)$$

成立, 则在  $k+1$  时刻均方误差达到最小。理想情况下, 梯度向量  $g_w(n)$  和矩阵  $R$  是已知的, 但实际情况却是我们只能得到梯度向量  $g_w(n)$  和矩阵  $R$  的估计值, 收敛因子  $\mu$  的作用就是为了避免采用  $g_w(n)$  和  $R$  的含有噪声的估计值而导致算法发散的问题。

下面直接给出 Newton-LMS 算法, 详细推导见文献[11].

$$\hat{R}^{-1}(n) = \frac{1}{1-\alpha} \left[ \hat{R}^{-1}(n-1) - \frac{\hat{R}(n-1)x(n)x^T(n)\hat{R}(n-1)}{\frac{1-\alpha}{\alpha} + x^T(n)\hat{R}^{-1}(n-1)x(n)} \right] \quad (2.48)$$

$$e(n) = d(n) - x^T(n)w(n) \quad (2.49)$$

$$w(n+1) = w(n) + 2\mu e(n) \hat{R}^{-1}(n)x(n) \quad (2.50)$$

式中  $\hat{R}^{-1}(n)$  是  $R^{-1}$  的估计值。

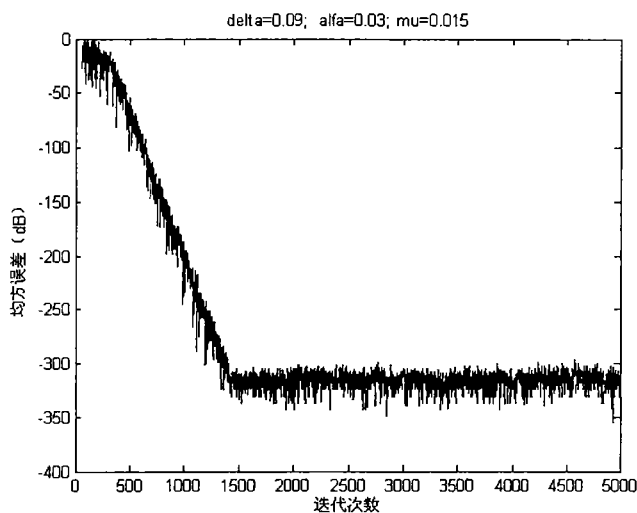


图 2.11

### 2.7.2 性能分析

从图 2.11 和图 2.12 看出, 在保持  $\delta$ 、 $\mu$  不变的情况下,  $\alpha$  的变化会对收敛速率产生很大的影响。当  $\alpha$  增大时, 收敛速度变慢。

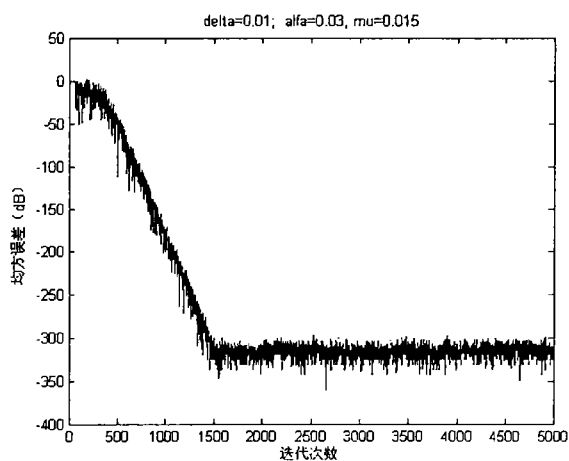


图 2.12

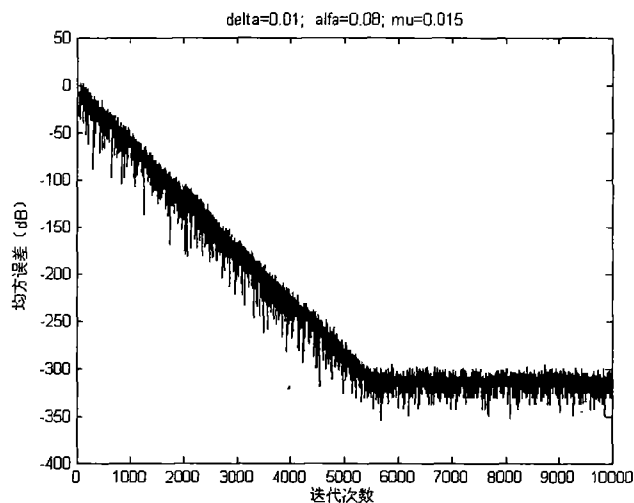


图 2.13

从图 2.12 和图 2.14 看出, 在保持  $\delta$ 、 $\alpha$  不变的情况下, 当  $\mu$  减小时, 收敛速度也变慢,  $\mu$  减小 0.005, 达到收敛所需的迭代次数从 1500 次增加到 3000 次。

从图 2.12 和图 2.13 看出, 在保持  $\alpha$ 、 $\mu$  不变的情况下,  $\delta$  的变化对算法没有明显的影响。

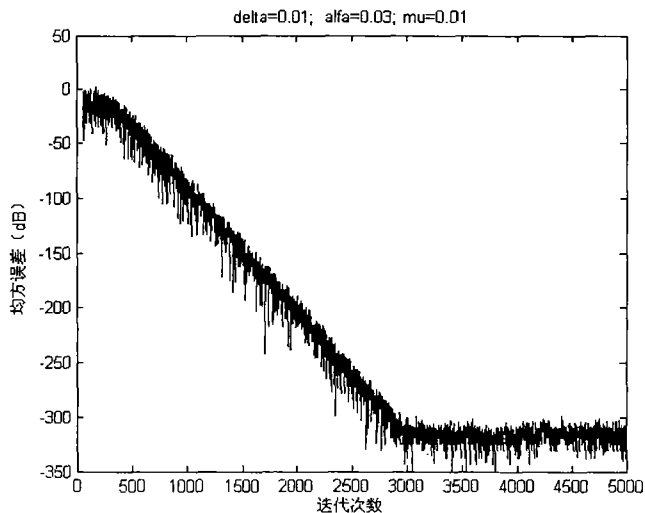


图 2.14



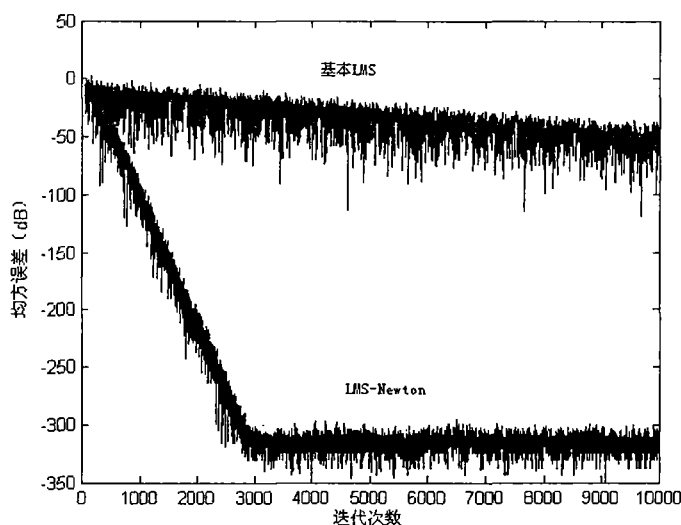


图 2.15

图 2.15 是在输入数据相关的情况下，基本 LMS 算法与 LMS-Newton 算法的收敛性的比较。从图中可以清楚地看出，当输入数据相关时，基本 LMS 算法的收敛速度将变得很慢，而 LMS-Newton 算法能很好地收敛。

## 2.8 小结

本章首先讨论、分析了基本 LMS 算法及其变型、量化误差 LMS 算法、解相关 LMS 算法、LMS-Newton 算法、并行延时 LMS 算法、自适应格型 LMS 算法的原理，然后简单地分析了各种算法的性能特点，同时还给出了算法实现的步骤。最后，利用 Matlab 对部分算法进行了仿真，给出了各种算法学习曲线的仿真结果，并对仿真结果进行了简单的说明。

### 3 LMS 算法的性能分析

#### 3.1 学习速率参数的选择

LMS 算法中的收敛因子  $\mu$  决定抽头权向量在每步迭代中的更新量, 是影响算法收敛速度的关键参数。由于 LMS 算法的目的是在更新过程中使抽头权向量逼近 Wiener 滤波器, 所以权向量的更新过程可以视为一种学习过程, 而  $\mu$  决定 LMS 算法学习过程的快慢, 因此我们把收敛因子  $\mu$  称为学习速率参数。基本 LMS 算法的收敛可分为均值收敛与均方收敛两种, 下面分别从这两种收敛的角度来讨论学习速率参数的选择问题。

##### 3.1.1 均值收敛的条件

由权向量更新公式:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu [d(n+1) - \mathbf{w}^H(n) \mathbf{u}(n+1)]^* \mathbf{u}(n+1)$$

我们可以建立基本 LMS 算法收敛必须满足的条件

$$E\{e(n)\} \rightarrow 0 \quad \text{当 } n \rightarrow \infty$$

或等价于  $\hat{\mathbf{w}}(n)$  的值收敛为最优 Wiener 滤波器:

$$\lim_{n \rightarrow \infty} E\{\hat{\mathbf{w}}(n)\} = \mathbf{w}_{opt}$$

对权向量更新公式两边取数学期望得

$$\begin{aligned} E\{\mathbf{w}(n)\} &= E\{\mathbf{w}(n-1)\} + \mu E\{e^*(n) \mathbf{u}(n)\} \\ &= E\{\mathbf{w}(n-1)\} + \mu E\{\mathbf{u}(n) [d(n) - \mathbf{u}^T \mathbf{w}^*(n-1)]^* \} \\ &= (\mathbf{I} - \mu \mathbf{R}) E\{\mathbf{w}(n-1)\} + \mu \mathbf{r} \end{aligned} \quad (3.1)$$

式中  $\mathbf{R} = E\{\mathbf{u}(n) \mathbf{u}^H(n)\}$  和  $\mathbf{r} = E\{\mathbf{u}(n) d^*(n)\}$

当  $k=1$  时

$$E\{\mathbf{w}(1)\} = (\mathbf{I} - \mu \mathbf{R}) E\{\mathbf{w}(0)\} + \mu \mathbf{r} \quad (3.2)$$

当  $k=2$  时, 利用 (3.1) 和 (3.2) 式, 则有

$$\begin{aligned}
 E\{w(2)\} &= (I - \mu R)E\{w(1)\} + \mu r \\
 &= (I - \mu R)^2 E\{w(0)\} + \mu \sum_{i=0}^1 (I - \mu R)^i r
 \end{aligned} \quad (3.3)$$

由 (3.1)、(3.2)、(3.3) 类推可得, 当  $k=n$  时,

$$E\{w(n)\} = (I - \mu R)^n E\{w(0)\} + \mu \sum_{i=0}^{n-1} (I - \mu R)^i r \quad (3.4)$$

令共轭对称的相关矩阵  $R$  的特征分解为

$$R = U \Sigma U^H \quad (3.5)$$

式中  $U$  为酉矩阵,  $\Sigma = \text{diag}(\lambda_1, \dots, \lambda_m)$  是对角阵, 其对角元素  $\lambda_i$  为矩阵  $R$  的特征值。

利用式 (3.5), 可以将式 (3.4) 改写为

$$E\{w(n)\} = (I - \mu U \Sigma U^H)^n E\{w(0)\} + \mu \sum_{i=0}^{n-1} (I - \mu U \Sigma U^H)^i r \quad (3.6)$$

$$\begin{aligned}
 \text{可以证明: } (I - \mu U \Sigma U^H)^i &= (U U^H - \mu U \Sigma U^H)^i \\
 &= [U(I - \mu \Sigma)U^H]^i = U(I - \mu \Sigma)U^H \dots U(I - \mu \Sigma)U^H \\
 &= U(I - \mu \Sigma)^i U^H
 \end{aligned} \quad (3.7)$$

若对所有特征值  $\lambda_i$  有  $|1 - \mu \lambda_i| < 1$ , 则  $\sum_{i=0}^{\infty} (I - \mu \Sigma)^i = (\mu \Sigma)^{-1}$ 。在此条件下,

式 (3.7) 可以改写成

$$\begin{aligned}
 \lim_{n \rightarrow \infty} \sum_{i=0}^{n-1} (I - \mu U \Sigma U^H)^i &= \sum_{i=0}^{\infty} U(I - \mu \Sigma)^i U^H \\
 &= U(I - \mu \Sigma)^{-1} U^H
 \end{aligned} \quad (3.8)$$

由于对角矩阵  $(I - \mu \Sigma)$  所有对角元素均小于 1, 故

$$\lim_{n \rightarrow \infty} (I - \mu \Sigma)^n = 0 \quad (3.9)$$

将式(3.7)、(3.8)和(3.9)代入式(3.6)，得到

$$\lim_{n \rightarrow \infty} E\{w(n)\} = 0 + U \Sigma^{-1} U^H r = R^{-1} r = w_{opt}$$

这表明权向量是均值收敛的。其条件是对角矩阵  $(I - \mu \Sigma)$  所有对角元素均小于 1，即

$$|1 - \mu \lambda_{\max}| < 1$$

解之得

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (3.10)$$

这就是权向量均值收敛时，学习速率参数  $\mu(n)$  必须满足此条件。

### 3.1.2 均方收敛的条件

若当迭代次数  $n$  趋于无穷大时，误差信号  $e(n) = d(n) - w^H(n-1)u(n)r$  的均方值收敛为一常数

$$\lim_{n \rightarrow \infty} E\{|e(n)|^2\} = C \quad C \text{ 为常数}$$

在实际应用中，相关矩阵  $R$  的最大特征值通常是未知的，因此式(3.10)的应用并不方便。为了克服这一困难，我们引入了 LMS 算法的均方收敛条件。

LMS 算法均方收敛的详细分析比其均值分析复杂得多。LMS 算法均方收敛的条件是：学习速率参数  $\mu$  满足不等式

$$0 < \mu < \frac{2}{\text{tr}[R]} \quad (3.11)$$

式中  $\text{tr}[R]$  是相关矩阵  $R$  的迹。详细证明见参考文献[4]。

由矩阵代数知

$$\text{tr}[\mathbf{R}] = \sum_{k=1}^M \lambda_k > \lambda_{\max} \quad (3.12)$$

故由式(3.10)、(3.11)和(3.12)得

$$0 < \mu < \frac{2}{\text{tr}[\mathbf{R}]} < \frac{2}{\lambda_{\max}} \quad (3.13)$$

这表明,若学习速率参数满足 LMS 算法均方收敛条件,则它必然满足 LMS 算法均值收敛条件。换句话说,若 LMS 算法是均方收敛的,则它必定也是均值收敛的。

根据矩阵代数,一个正方形的迹等于它的对角元素之和。当横向滤波器是空域滤波器时,其  $M$  个输入端的输入信号分别是  $M$  个传感器的观测数据。此时,空域滤波器的输入信号向量为  $\mathbf{u}(n) = [u_1, \dots, u_M]^T$ 。由于相关矩阵  $\mathbf{R} = E\{\mathbf{u}(n)\mathbf{u}^H(n)\}$  的第  $i$  个对角元素为  $R_i(0) = E\{u_i(n)u_i^*(n)\}$  是输入信号  $u_i(n)$  的均方值或能量。因此,相关矩阵  $\mathbf{R}$  的迹等于在滤波  $M$  个输入端上测得的总的输入能量,即

$$\text{tr}[\mathbf{R}] = \sum_{i=1}^M R_i(0) = \sum_{i=1}^M E\{|u_i|^2\} = \text{总的输入能量}$$

这表明, LMS 算法均方收敛的条件也可以写成

$$0 < \mu < \frac{2}{\text{总的输入能量}} \quad (3.14)$$

### 3.1.3 自适应学习速率参数

前两节从 LMS 算法的均值收敛和均方收敛的角度分别得到了学习速率参数应该满足的条件。在 LMS 算法中最简单的学习速率参数选择是取  $\mu(n)$  为一常数,即

$$\mu(n) = \mu$$

式中  $\mu$  由式(3.13)和(3.14)确定。然而,这种方法会引起收敛与稳态性能的矛

盾，因为大的学习速率能够提高滤波器的收敛速度，但稳态性能会降低，反之，

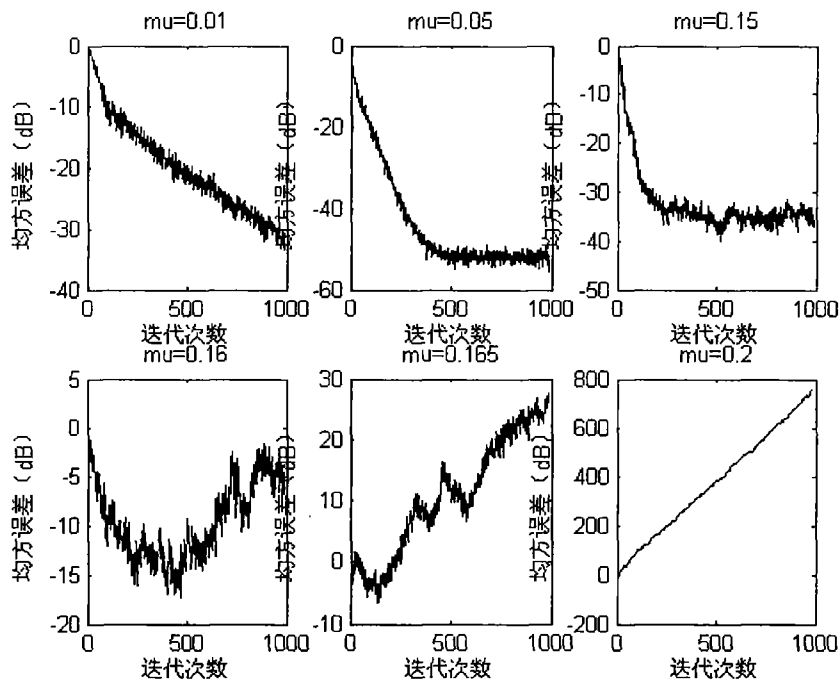


图 3.1  $\mu$  对收敛的影响

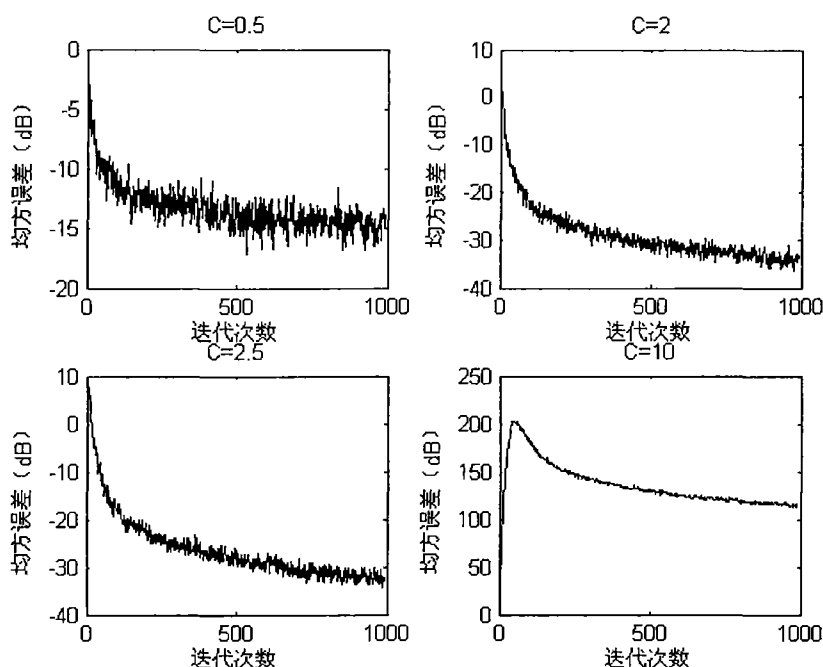
为了提高稳态性能而采用小的学习速率时，收敛速度会变慢。所以学习速率的选择应该兼顾稳态性能与收敛速度。简单而有效的方法就是在不同的迭代时间使用不同的学习速率参数，即采用时变的学习速率。

时变学习速率的思想最早可追溯到 Robbins 和 Monro 于 1951 年提出的随机逼近法。最简单的时变学习速率为

$$\mu(n) = \frac{c}{n}$$

式中  $c$  为一常数。这种选择常称模拟退火法则。需要注意的是，当参数  $c$  选择比较大时，LMS 算法经过若干次迭代后即陷于发散。

更好的方法是在暂态（过渡）阶段使用大的学习速率，而在稳态使用小的学习速率。学习速率参数的这种选择方法称为“换档变速方法”。“固定+时变”的学习速率就是典型的换档变速方法。下面举两个典型例子。


 图 3.2 模拟退火法则中  $C$  对收敛的影响

第一个例子是使用“先搜索，后收敛”法则

$$\mu(n) = \frac{\mu_0}{1 + (n/\tau)} \quad (3.15)$$

式中  $\mu_0$  为一固定的学习速率参数，而  $\tau$  表示搜索时间常数。由 (3.15) 可以看出，这种法则在  $n \leq \tau$  的迭代时间内使用近似固定的学习速率  $\mu_0$ ；而当迭代时间  $n$  比搜索时间参数  $\tau$  大时，学习速率则随时间衰减，并且衰减速度越来越快。

第二个例子是“先固定，后指数衰减”的法则

$$\mu(n) = \begin{cases} \mu_0, & n \leq N_0 \\ \mu_0 e^{-N_d(n-N_0)}, & n \geq N_0 \end{cases} \quad (3.16)$$

式中  $\mu_0$  和  $N_d$  分别为正的常数， $N_0$  为正整数。

上述时变的学习速率是预先确定的，与 LMS 算法的实际运行状态并没有直接的关系。如果时变的学习速率是由 LMS 算法的实际运行状态控制的，则这类

时变的学习速率参数称为自适应学习速率,也称“学习规则的学习”,它是 Amari 于 1967 年提出的。到现在已经提出许多方法选择自适应学习速率。下面介绍三个例子:

① 通过检验 LMS 算法估计误差相邻样本值的极性控制学习速率。若估计误差存  $m_0$  个相邻的符号变化,则适当减小学习速率;如果存在  $m_1$  个相邻的同符号,则适当增加学习速率。

② 根据预测误差的平方来调节学习速率。

③ 以上方法需要用户选择一些附加的常数和初始学习速率,它们都是根据“初始阶段使用大的学习速率,稳态阶段使用小的学习速率”的语言规则,并把这些语言规则转换成数学模型进行学习速率参数的调节。另一方面,学习速率的调节也可以使用模糊系统理论和语言模型来实现,构成所谓的模糊步长调节。

## 3.2 LMS 算法的统计性能

下面将用独立性理论来对 LMS 算法的统计性能进行分析。LMS 算法的独立性理论最早是由 Widrow 和 Mazo 提出的,它的核心是下面 4 个独立性假设:

- ① 输入向量  $\mathbf{u}(1), \mathbf{u}(2), \dots, \mathbf{u}(n)$  相互统计独立;
- ② 在时刻  $n$ , 输入向量  $\mathbf{u}(n)$  与所有过去所有时刻的期望响应  $d(1), d(2), \dots, d(n-1)$  统计独立;
- ③ 时刻  $n$  的期望响应  $d(n)$  与  $n$  时刻的输入向量  $\mathbf{u}(n)$  相关,但与过去时刻的输入向量统计独立;
- ④ 输入向量  $\mathbf{u}(n)$  和期望响应  $d(n)$  对所有  $n$  组成联合高斯分布的随机变量。

### 3.2.1 稳态剩余输出均方误差

令  $w_{opt}$  表示最优 Wiener 滤波器的权值,则权误差向量定义为

$$\boldsymbol{\varepsilon}(n) = \mathbf{w}(n) - \mathbf{w}_{opt} \quad (3.17)$$



于是, 由 LMS 算法产生的估计误差可以定义为

$$\begin{aligned} e(n) &= d(n) - \mathbf{w}^H(n) \mathbf{u}(n) \\ &= d(n) - \mathbf{w}_{opt}^H \mathbf{u}(n) - \boldsymbol{\varepsilon}^H(n) \mathbf{u}(n) \\ &= e_{opt}(n) - \boldsymbol{\varepsilon}^H(n) \mathbf{u}(n) \end{aligned} \quad (3.18)$$

式中  $e_{opt}(n)$  是最优 Wiener 滤波器的估计误差。考虑抽头权向量  $\mathbf{w}(n)$  的估计误差的均方值, 简称均方误差

$$\xi(n) = \text{MSE}(\mathbf{w}(n)) = E\{|e(n)|^2\} \quad (3.19)$$

利用独立性假设得

$$\begin{aligned} \xi(n) &= E\{[e_{opt}(n) - \boldsymbol{\varepsilon}^H(n) \mathbf{u}(n)][e_{opt}^*(n) - e_{opt}^H(n) \boldsymbol{\varepsilon}(n)]\} \\ &= \xi_{min}(n) + E\{\boldsymbol{\varepsilon}^H(n) \mathbf{u}(n) \mathbf{u}^H(n) \boldsymbol{\varepsilon}(n)\} \end{aligned} \quad (3.20)$$

式中

$$\xi(n) = \text{MSE}(\mathbf{w}(n)) = E\{|e(n)|^2\} = E\{e_{opt}(n) e_{opt}^*(n)\} \quad (3.21)$$

是最优 Wiener 滤波器产生的最小均方误差。

计算式 (3.20) 右边的第二项, 有

$$\begin{aligned} E\{\boldsymbol{\varepsilon}^H(n) \mathbf{u}(n) \mathbf{u}^H(n) \boldsymbol{\varepsilon}(n)\} &= E\{\text{tr}[\boldsymbol{\varepsilon}^H(n) \mathbf{u}(n) \mathbf{u}^H(n) \boldsymbol{\varepsilon}(n)]\} \\ &= E\{\text{tr}[\mathbf{u}(n) \mathbf{u}^H(n) \boldsymbol{\varepsilon}(n) \boldsymbol{\varepsilon}^H(n)]\} \\ &= \text{tr}\{E[\mathbf{u}(n) \mathbf{u}^H(n) \boldsymbol{\varepsilon}(n) \boldsymbol{\varepsilon}^H(n)]\} \end{aligned} \quad (3.22)$$

式中使用了矩阵迹的性质  $\text{tr}[\mathbf{AB}] = \text{tr}[\mathbf{BA}]$ , 并假定  $\mathbf{u}(n)$  与  $\boldsymbol{\varepsilon}(n)$  统计独立。

利用独立性假设, 式 (3.22) 又可以写作

$$\begin{aligned} E\{\boldsymbol{\varepsilon}^H(n) \mathbf{u}(n) \mathbf{u}^H(n) \boldsymbol{\varepsilon}(n)\} &= \text{tr}\{E[\mathbf{u}(n) \mathbf{u}^H(n)] E[\boldsymbol{\varepsilon}(n) \boldsymbol{\varepsilon}^H(n)]\} \\ &= \text{tr}[\mathbf{RK}(n)] \end{aligned} \quad (3.23)$$

式中  $\mathbf{R} = E\{\mathbf{u}(n) \mathbf{u}^H(n)\}$  是输入向量的相关矩阵, 而  $\mathbf{K}(n) = E\{\boldsymbol{\varepsilon}(n) \boldsymbol{\varepsilon}^H(n)\}$  表示  $n$  时刻的滤波器权误差向量的相关矩阵, 简称权误差相关矩阵。

将式(3.23)代入式(3.20)即可将 LMS 算法中的均方误差表示为

$$\xi(n) = \xi_{\min}(n) + \text{tr}[\mathbf{RK}(n)] \quad (3.24)$$

由  $n$  时刻的自适应算法产生的均方误差  $\xi(n)$  与由最优 Wiener 滤波器产生的最小均方误差  $\xi_{\min}(n)$  之差称为  $n$  时刻的自适应算法的剩余均方误差, 记作

$$\xi_{\text{ex}}(n) = \xi(n) - \xi_{\min} = \text{tr}[\mathbf{RK}(n)] \quad (3.25)$$

当  $n$  趋于无穷大时, 剩余均方误差的极限值称为稳态剩余均方误差 (或称渐远剩余均方误差), 记作

$$\xi_{\text{ex}}(n) = \xi_{\text{ex}}(\infty) = \lim_{n \rightarrow \infty} \text{tr}[\mathbf{RK}(n)] \quad (3.26)$$

### 3.2.2 稳态剩余输出能量

我们考虑一种特殊情况: 当期望响应  $d(n) \equiv 0$  时, 最小均方误差 (MMSE) 准则的代价函数

$$\nabla J(n) = [\nabla_0 J(n), \nabla_1 J(n), \dots, \nabla_{M-1} J(n)]^T$$

变作

$$J(n) = E\{|\mathbf{w}^H \mathbf{u}(n)|^2\} \quad (3.27)$$

由于上式右边代表滤波器输出的能量, 所以上式的最小化称为最小输出能量 (MOE) 准则。

仿照均误差的定义, 我们可以定义时刻滤波器抽头权向量  $\mathbf{w}(n)$  的平均输出能量

$$\eta(n) = \text{MOE}(\mathbf{w}(n)) = E\{|\mathbf{w}^H \mathbf{u}(n)|^2\} \quad (3.28)$$

由于权误差向量  $\boldsymbol{\varepsilon}(n) = \mathbf{w}(n) - \mathbf{w}_{\text{opt}}$ , 并且  $\mathbf{w}_{\text{opt}}$  与滤波器输入向量  $\mathbf{u}(n)$  统计独立, 故由式(3.28)得

$$\eta(n) = E\{|\mathbf{w}_{\text{opt}} + \boldsymbol{\varepsilon}(n)|^H \mathbf{u}(n)|^2\}$$

$$\begin{aligned}
 &= E\{|\mathbf{w}_{opt}^H \mathbf{u}(n)|^2\} + E\{\varepsilon^H(n) \mathbf{u}(n) \mathbf{u}^H(n) \varepsilon(n)\} \\
 &= \eta_{min} + E\{\varepsilon^H(n) \mathbf{u}(n) \mathbf{u}^H(n) \varepsilon(n)\} \quad (3.29)
 \end{aligned}$$

式中  $\eta_{min} = E\{|\mathbf{w}_{opt}^H \mathbf{u}(n)|^2\}$  表示最优滤波器的输出能量，它就是自适应滤波器所能达到的最小输出能量。我们定义输出能量

$$\eta_{ex}(n) = \eta(n) - \eta_{min} \quad (3.30)$$

并利用式(3.23)有

$$\eta_{ex}(n) = E\{\varepsilon^H(n) \mathbf{u}(n) \mathbf{u}^H(n) \varepsilon(n)\} = \text{tr}[\mathbf{R}\mathbf{K}(n)] \quad (3.31)$$

比较式(3.31)与式(3.25)知

$$\eta_{ex}(\infty) = \xi_{ex}(\infty) \quad (3.32)$$

从上式我们可以看出，滤波器的稳态剩余输出能量与稳态剩余输出均方误差等价。

以上分析表明，尽管根据 MMSE 准则与 MOE 准则设计的滤波器抽头权向量可能不同，但它们的稳态剩余均方误差和稳态剩余输出能量等价。

### 3.3 LMS 算法的跟踪性能

上面对 LMS 算法统计性能的分析是在 Wiener 滤波器固定不变的基本假设下进行的，这些统计性能是基本 LMS 算法所具有的平均性能，它们适合于平稳环境。

在非平稳环境下，系统的参数是时变的，因而 Wiener 滤波器的参数也应该是时变的，以跟踪系统的动态变化。评价 LMS 算法对非平稳环境的适应能力的指标是 LMS 算法的跟踪性能。根据参数随时间变化的快慢，时变系统有快时变和慢时变之分，这里只研究慢时变。

一个未知的动态系统可以用一横向滤波器建模，该滤波器的抽头权向量  $\mathbf{w}_{opt}$  服从一阶马尔可夫过程：

$$w_{opt}(n+1) = \alpha w_{opt}(n) + \omega(n) \quad (3.33)$$

式中  $\alpha$  为一固定参数, 对于慢时变系统,  $\alpha$  是一个非常接近 1 的正数;  $\omega(n)$  为过程噪声, 其均值为 0, 相关矩阵为  $\mathbf{Q}$ 。

横向滤波器的输出  $w_{opt}^H(n)\mathbf{u}(n)$  逼近期望响应, 其逼近误差  $v(n)$  称为测量噪声。因此, 横向滤波器的期望响应可以表示为

$$d(n) = w_{opt}^H(n)\mathbf{u}(n) + v(n) \quad (3.34)$$

对滤波器的输入、过程噪声和测量噪声作如下假设:

- ① 过程噪声向量  $\omega(n)$  与输入向量  $\mathbf{u}(n)$ 、测量噪声向量  $v(n)$  独立;
- ② 输入向量  $\mathbf{u}(n)$  和测量噪声  $v(n)$  相互独立;
- ③ 测量噪声  $v(n)$  为白噪声, 它具有零均值和有限方差  $\sigma_v^2 < \infty$ 。

### 3.3.1 非平稳度 $\alpha$

为描述模型的快、慢变化, Macchi 将时变系统的非平稳度定义为过程噪声向量  $\omega(n)$  引起的平均噪声功率与测量噪声引起的平均噪声功率之比, 即

$$\alpha = \left( \frac{E\{|\omega^H(n)\mathbf{u}(n)|^2\}}{E\{|v(n)|^2\}} \right)^{1/2} \quad (3.35)$$

非平稳度  $\alpha$  只是一个时变系统的特征描述, 它并不对自适应滤波器作任何描述。

利用过程噪声向量  $\omega(n)$  与输入向量  $\mathbf{u}(n)$  之间的统计独立性, 并注意到对于标量  $x^H y$  而言,  $E\{x^H y\} = \text{tr}[E\{x^H y\}]$  容易得到式 (3.35) 的分子为

$$\begin{aligned} E\{|\omega^H(n)\mathbf{u}(n)|^2\} &= E\{\omega^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\omega(n)\} \\ &= E\{\text{tr}[\omega^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\omega(n)]\} \\ &= \text{tr}[E\{\omega(n)\omega^H(n)\mathbf{u}(n)\mathbf{u}^H(n)\}] \\ &= \text{tr}[E\{\omega(n)\omega^H(n)\}E\{\mathbf{u}(n)\mathbf{u}^H(n)\}] \\ &= \text{tr}[\mathbf{QR}] \end{aligned} \quad (3.36)$$

式中  $\mathbf{Q} = E\{\omega(n)\omega^H(n)\}$  是过程噪声向量  $\omega(n)$  的相关矩阵;  $\mathbf{R} = E\{\mathbf{u}(n)\mathbf{u}^H(n)\}$  表示

输入向量  $\mathbf{u}(n)$  的相关矩阵。

另一方面，式 (3.35) 的分母是零均值的测量噪声  $\mathbf{V}(n)$  的方差  $\sigma_v^2$ 。将这一结果和式 (3.36) 代入式 (3.35)，即可将非平稳度简写为

$$\alpha = \frac{1}{\sigma_v} (\text{tr}[\mathbf{Q}\mathbf{R}])^{1/2} = \frac{1}{\sigma_v} (\text{tr}[\mathbf{R}\mathbf{Q}])^{1/2} \quad (3.37)$$

式中  $\text{tr}[\mathbf{Q}\mathbf{R}] = \text{tr}[\mathbf{R}\mathbf{Q}]$  是因为矩阵乘积  $\mathbf{Q}\mathbf{R}$  和  $\mathbf{R}\mathbf{Q}$  具有相同的对角线元素。

### 3.3.2 失调 M

除了前面介绍过的收敛因子外，失调 (misadjustment) 是衡量自适应滤波器性能的另一个重要测度。自适应滤波器的失调定义为滤波器稳态剩余均方误差  $\xi_{ex}$  与滤波器最小均方误差  $\xi_{min}$  之比，即

$$M = \frac{\xi_{ex}}{\xi_{min}} = \frac{\mu(N+1)\sigma_x^2}{1 - \mu(N+1)\sigma_x^2} \quad (3.38)$$

式中，稳态剩余均方误差  $\xi_{ex}$  定义为滤波器输出的实际均方误差与最小均方误差之差，即  $\xi_{ex} = \xi_{out} - \xi_{min}$ 。显然，当  $\xi_{ex} = 0$  时，滤波器的输出刚好达到最小输出均方误差，因此，它是最小均方误差意义下的最优滤波器，此时，失调  $M = 0$ ，即滤波器不存在任何失调。由此可以看出，失调  $M$  实际上是一个衡量滤波器偏离最优滤波器的测度。只要剩余输出能量不等于零，便称滤波器存在失调。通常希望自适应滤波器的失调越小越好，这取决于滤波器的设计和滤波器所处的环境。

表 3.1 是引用文献 [11] 的分析结果 (详细分析见文献 [11] P<sub>69</sub>-P<sub>72</sub> 和 P<sub>107</sub>-P<sub>108</sub>)，利用失调  $M$  对基本 LMS 算法和变换域解相关 LMS 算法性能进行比较。表中  $\frac{\lambda_{max}}{\lambda_{min}}$  为输入相关矩阵的特征扩展值，对比表中  $M$  数据：当特征扩展值较大时，变换域解相关 LMS 算法具有比基本 LMS 算法更好的性能。

需要注意的是，只有当采用的变换是有效的时候，变换域 LMS 算法的收敛性能才能提高。

基本 LMS 算法			变换域解相关 LMS 算法		
$\mu$	$\frac{\lambda_{\max}}{\lambda_{\min}}$	失调 M	$\mu$	$\frac{\lambda_{\max}}{\lambda_{\min}}$	失调 M
0.02	1	0.2027	0.01	1	0.2027
0.01280	20	0.1298	0.0066	20	0.2037
0.01024	80	0.1045	0.0051	80	0.2093

表 3.1 基本 LMS 算法与变换域解相关 LMS 算法的对比

### 3.3.3 非平稳度 $\alpha$ 与失调 M 之间的关系

对于一个由 MMSE 准则设计的滤波器来说，其最小均方误差  $J_{\min}$  等于测量噪声的方差，即有

$$\xi_{\min} = \sigma_v^2 \quad (3.39)$$

另一方面，由马尔可夫模型公式(3.33)可以看出，过程噪声向量  $\omega(n)$  实际上就是滤波器权误差向量，即  $\omega(n) \approx w_{opt}(n+1) - w_{opt}(n) = \varepsilon(n)$ ，因为式(3.33)中的系数  $\alpha$  非常接近 1，这表明  $n$  时刻的过程噪声向量  $\omega(n)$  的相关矩阵  $\mathbf{Q}(n) = E\{\omega(n)\omega^H(n)\} \approx E\{\varepsilon(n)\varepsilon^H(n)\} = \mathbf{K}(n)$ ，将这一关系代入式(3.31)，得到自适应滤波器  $n$  时刻的剩余输出能量

$$\eta_{ex}(n) \approx \text{tr}[\mathbf{RQ}(n)] \quad (3.40)$$

若自适应滤波器不存在失调，则其稳态剩余输出能量  $\xi_{ex} = \eta_{ex}(\infty) \approx \text{tr}[\mathbf{RQ}(n)]$ ，即最小均方误差滤波器与最小能量滤波器等价。若自适应滤波器存在失调，其稳态剩余输出能量

$$\xi_{ex} > \text{tr}[\mathbf{RQ}(n)] \quad (3.41)$$

将式(3.39)和式(3.41)代入式(3.38)有

$$M \geq \frac{\text{tr}[RQ]}{\sigma_v^2} = \alpha^2 \quad (3.42)$$

换句话说，自适应滤波器的失调  $M$  是时变系统的非平稳度的平方值的上界。

从上面的分析我们可以得出以下两点结论：

① 对于慢时变系统，由于其非平稳度  $\alpha$  小，因此自适应滤波器可以跟踪时变系统的变化。

② 若时变系统的变化太快，以便致于非平稳度  $\alpha$  大于 1，那么在这样的情况下，由自适应滤波器造成的失调  $M > 1$ ，即失调将超过 100%，这意味着自适应滤波器将不可能跟踪这种快时变系统的变化。

### 3.4 影响 LMS 算法的几个因素

LMS 算法在自适应滤波算法中使用最广泛，但几个因素也影响和制约着它的发展。这些因素包括不稳定性的影响、噪声输入端信号的影响、有限字长的影响和系数漂移的影响等，下面将分别讨论这些因素对 LMS 算法的影响。

#### 3.4.1 不稳定性的影响

当滤波器的工作环境是平稳的情况下，如果权重系数收敛到一个值后，信号的统计特性（信号的平均值、方差、自相关等）发生变化，那么只要这个改变低得还能使滤波器收敛，滤波器就能重新调整权重系数到一组新的最佳值。误差性能曲面的形状和取向都不变，自适应滤波器收敛和工作在最佳点附近。

在实际情况下，自适应滤波器所处的环境可能是非平稳的，即信号的统计特性不断改变时，系统的性能曲面的底部和最小值都在不断变动之中，它的形状和取向也可能不断变动，这时算法不仅要求得曲面的最小点，还要不断跟踪它的位置变动，这会使滤波器性能严重下降。

#### 3.4.2 噪声输入端的信号的影响

算法的性能还取决于噪声输入端的信号，而受所希望提取的信号输入端的

信号的影响很弱。在某些应用中，噪声输入端的信号中除了噪声之外，也包括

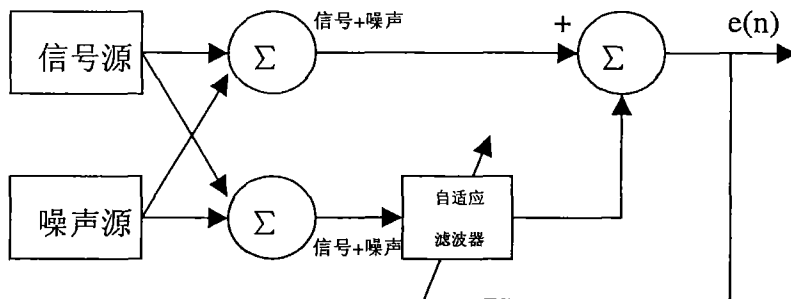


图 3.3 输入端有信号和噪声的自适应抵消器

低电平信号，如图 3.3 所示。这样，运算将抵消掉一部分有用的信号分量。这时，自适应滤波对信噪比仍然有相当程度的改善，只是产生小信号失真，但是如果噪声输入端的信号中只包含有用信号而不包含噪声分量，则输出就完全消失。

### 3.4.3 有限字长的影响

前面我们在分析算法的性能时没有考虑到算法的精度问题，分析结果是在假设“无限精度”（即字长很长）时得到的。然而处理器能处理的数据位数（字长）是有限的，而且为满足速度需要，要求利用短字长实现自适应滤波算法，这就使得 LMS 算法的特性与用无限精度时有很大不同。LMS 算法的递归性质使得运算过程中  $e(n)$  和  $w(n)$  的长度不断增加，但因字长有限，所以多出的位不断被舍弃，这导致了有限字长对 LMS 滤波的影响。具体有以下三个方面：

1、可能导致自适应滤波器无法收敛到它的最佳解。

图 3.4(b) 是将图 3.4(a) 中的迭代次数 500–600 之间的图截取后放大得到的。从图 3.4(b) 中我们可以看到，滤波器的误差最后并没有收敛到一个值上，而是大致以  $-49.5\text{dB}$  为中心线振荡。产生这种振荡的原因是因为我们进行计算时采用的字长（数据位数）有限，计算时超出所用位数的数被舍入或截断，由此产生了舍入或截尾误差。同等条件下，舍入误差优于截尾误差，舍入或截尾误



差可通过增加数据的位数来减小。

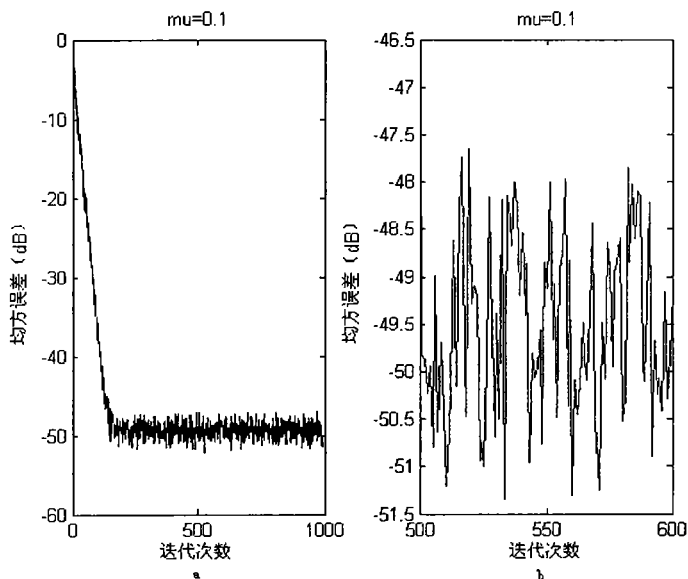


图 3.4

## 2、滤波器输出中可能包括随机噪声。

当对信号进行量化时，由于精度的限制，会产生量化误差，由此形成白噪声。此外，在运算中，若数据溢出时不采用饱和处理或对数据舍入、截尾也会形成噪声。

要降低随机噪声对信号的影响，最简单的办法是提高信噪比。提高信噪比的方法，一种是增大输入信号，另一种是增加字长。字长每增加一位，信噪比约增加 6dB。

## 3、可能导致运算过早中止。

理论上，当收敛因子  $\mu$  趋于零时，通常最小均方误差也达到稳定状态，但由于量化误差的影响，当  $\mu$  小于某个值时，均方误差将显著增加，而且算法可能停止某些滤波器系数的更新。文献[11]的附录 A 中对 LMS 算法中有限字长（量化效应）进行了详细分析。

实用中，对低阶滤波器（100 个系数以内）使用 16 位字长的系数就足以保证  $16 \times 16$  位乘法和 32 位累加运算的性能。

### 3.4.4 系数漂移的影响

对某些特殊的输入，递归运算会使得滤波器系数从最佳值缓慢漂移，最终超出允许的字长，这个问题中 LMS 算法的内在缺陷，它导致滤波器长期性能的退化。为解决此问题，我们引入一个小的泄漏因子  $\delta$  将系数略微推向 0，以此来抵消系数漂移。下面是两种设置  $\delta$  的方法：

$$1、w_{k+1}(n) = \delta w_k(n) + 2\mu e_k x_{k-n} \quad 0 < \delta < 1$$

$$2、w_{k+1}(n) = w_k(n) + 2\mu e_k x_{k-n} \pm \delta \quad 0 < \delta < 1$$

这两种设置  $\delta$  的方法抵消了系数漂移，使漂移得到抑制，但同时也在  $e_k$  中引入了误差项。

### 3.5 小结

本章对 LMS 算法的统计性能、跟踪性能以及学习速率参数的选择进行了分析，讨论了均值收敛和均方收敛的条件，推导了稳态剩余输出均方误差和稳态剩余输出能量的公式，还对影响跟踪性能的两个参数：非平稳度  $\alpha$  和失调  $M$  进行了分析说明，并讨论了非平稳度  $\alpha$  与失调  $M$  之间的关系。最后对影响 LMS 算法性能的几个因素分别进行讨论、分析。

## 4 LMS 算法在励磁电源控制系统中的应用

### 4.1 概述

在励磁电源控制系统的设计过程中,最先我们设计用 51 单片机来实现对系统的控制,但经实践发现 51 单片机无法满足此系统的要求。系统要求输出电流范围为  $0\sim 500\text{A}$ , 正常工作时为  $200\text{A}\sim 300\text{A}$ , 当负载不变时, 输出电流误差不得超过  $0.1\%$ 。现场干扰源很多, 电源电压变化很大, 特别是在负载变化时, 加之励磁模块产生的干扰, 使 51 单片机不能正常工作。因干扰源与 51 单片机控制电路板相距很近甚至有的直接与控制电路板相连(励磁模块), 用屏蔽和硬件滤波无法完全解决干扰问题。后来采用在 51 单片机内部进行软件滤波, 但因 51 单片机的数据处理能力和处理速度不够, 导致输出振荡, 无法稳定在设定值上。

经过论证, 我们决定采用 DSP 来进行此系统的设计, 利用 DSP 在数据处理能力和处理速度方面的优势来实现软件滤波功能, 并结合 PID 控制方法对励磁电源模块进行控制。

### 4.2 系统硬件设计

本文的硬件电路的开发环境是 Windows98 下的 Protel 99se, 电路原理图的绘制和电路板的印制都是利用 Protel 99se 中相应的功能完成的。

#### 4.2.1 硬件设计思路

首先将传感器送来的信号进行一次硬件低通滤波, 然后送入变送电路将其变为  $0\sim 10\text{V}$  的电压, 以便进行 A/D 转换。接着将经过变送电路的信号送入 A/D 转换器进行 A/D 转换, 经 A/D 转换后的数据直接送入 DSP 中进行软件滤波、数字 PID 等处理, 处理后的结果送至 D/A 转换器, 最后用 D/A 转换的输出去控制励磁电源模块。

#### 4.2.2 TMS320VC5410 简介

在本项目中，考虑到成本、开发时间及元器件采购，决定采用德州仪器的定点 DSP 芯片 TMS320VC5410。此芯片从数据处理能力、处理速度和运算精度上来说都能满足我们的要求。

TMS320VC5410 是采用了改进的哈佛结构的定点 DSP 处理器，它集成了丰富的硬件逻辑和外部接口资源，在提高性能的同时，降低了成本和体积。

它有一条程序存储器总线和三条数据存储器总线、一个高度并行的 40 位算术逻辑单元 (ALU)、特殊应用硬件逻辑、在片内存和额外的在片外设。它还有一套使操作灵活和加快 DSP 运行的专用指令。分离的程序和数据空间可同时并行进行程序指令和数据的存取，一个周期内可完成两次读和一次写操作。并行存储指令和特殊的专用指令使增强的哈佛结构得到充分利用。此外，数据能在数据和程序空间中进行传送。这种并行性还支持能在单个机器周期内完成的功能强大的算术、逻辑和位操作运算。同时，5410 还包括中断管理、重复操作和功能调用等控制机制。总的存储器空间为  $192\text{k} \times 16\text{bit}$ ，分为 3 个部分： $64\text{k} \times 16\text{bit}$  的数据存储空间、 $64\text{k} \times 16\text{bit}$  的程序存储空间、 $64\text{k} \times 16\text{bit}$  的并行 I/O 空间。

TMS320VC5410 有以下主要特点：

- 操作性能高达 100MIPS；
- 整合维特比操作；
- 备有三个掉电模式，便于用电池供电的应用；
- 整合随机存储器和只读存储器配置；
- 40bit 加法器和两个 40bit 累加器用于支持并行指令；
- 40bit 算术逻辑部件和双 16bit 配置性能，用于双重单周期操作；
- 4 个内部总线和双地址生成器，以进行多重操作数运算来减少存储器瓶颈现象；
- $17 \times 17$  乘法器，允许 16bit 带符号或不带符号的乘法；

- 单周期正规化及指数译码；
- 8 个辅助寄存器和 1 个软件栈，允许使用 C 语言编译器。

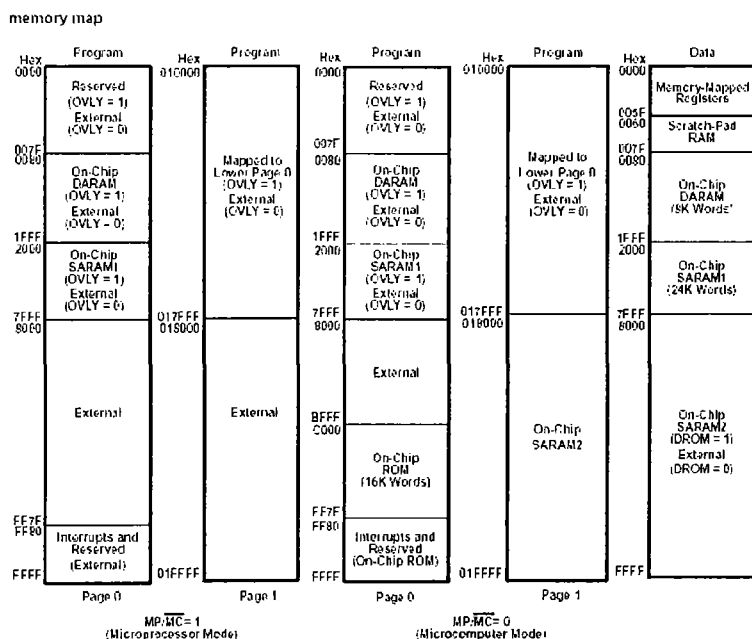


图 4.1 TMS320VC5410 存储器映射

图 4.1 给出了 TMS320VC5410 的存储器映射图。由图可知 TMS320VC5410 有 64K×16bit 的片内 RAM，16K×16bit 的片内 ROM。

### 4.2.3 A/D、D/A 转换芯片简介

本项目中，综合考虑转换精度、可靠性、价格等因素后，A/D 芯片采用美国模拟器件公司的工业级 12 位 A/D 转换芯片 AD1674AD 芯片，D/A 芯片采用美国国家半导体公司的工业级 12 位 A/D 转换芯片 DAC1208LIN。

#### 1、AD1674 简介

AD1674 是美国模拟数字公司推出的单片高速 12 位逐次比较型 A/D 转换器，它是由 AD574A 改进而来的，内部比 AD574A 多了采样保持器。内置双极性电路构成的混合集成转换芯片，具有外接元件少，功耗低，精度高等特点，并且具有自动校零和自动极性转换功能。图 4.2 和图 4.3 分别给出了 AD1674 的引脚功

能和内部功能框图。表 4.1 给出了 AD1674 各控制端的标志意义。

其主要功能特性如下：

- 分辨率：12 位
- 非线性误差：小于  $\pm 1/2\text{LBS}$  或  $\pm 1\text{LBS}$
- 转换速率： $10\mu\text{s}$
- 模拟电压输入范围：两档四种  
 $0-10\text{V}$  和  $0-20\text{V}$ ,  $0-\pm 5\text{V}$  和  $0-\pm 10\text{V}$
- 电源电压： $\pm 15\text{V}$  和  $5\text{V}$
- 数据输出格式：12 位/8 位
- 芯片工作模式：全速工作模式和单一工作模式

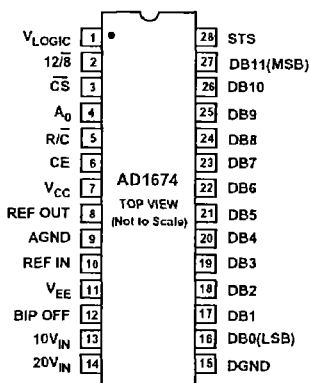


图 4.2 AD1674 引脚图

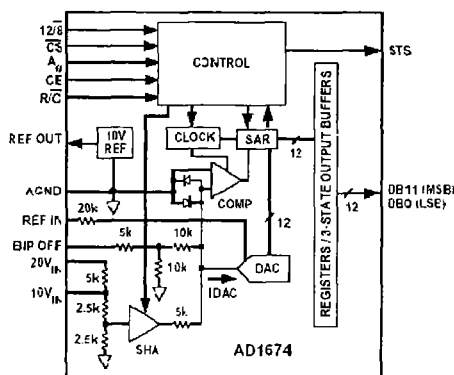


图 4.3 AD1674 内部功能框图

CE	$\overline{CS}$	$R/\overline{C}$	$12/\overline{8}$	A0	工作状态
0	X	X	X	X	禁止
x	1	X	X	X	禁止
1	0	0	X	0	启动 12 位转换
1	0	0	X	1	启动 8 位转换
1	0	1	接+5V	X	12 位并行输出有效
1	0	1	接 0V	0	高 8 位并行输出有效
1	0	1	接 0V	1	低 4 位并行输出有效

表 4.1 AD1674A 控制端标志意义

## 2、DAC1208 简介

DAC1208 芯片是美国国家半导体公司生产的工业级 12 位分辨率的乘法 D/A 转换芯片，这种芯片采用双缓冲结构，转换精度较高，价格低廉，接口简单，得到广泛应用。其内部有由 8 位寄存器和 4 位寄存器组成的输入寄存器以及 12 位 DAC 寄存器和 12 位乘法 DAC。模拟执行部件（12 位乘法 DAC）由 R-2R 梯形结构和 12 个 CMOS 电流开关组成。由于 DAC1208 是 12 位数据总线，能直接与 16 位的 DSP 芯片数据总线相连。

其主要功能特性如下：

- 分辨率：12 位；
- 逻辑电平与 TTL 电平兼容；
- 输出电流稳定时间：1  $\mu$ s；
- 电流输出：最大输出电流 2.5mA；
- 单电源供电：+5~+15V；
- 数据输入格式：可单缓冲、双缓冲或直接数字输入；
- 只需在满量程下调整线性度；

图 4.4 和图 4.5 分别给出了 DAC1208 的引脚功能和内部功能框图。





#### 4.2.4 系统框图

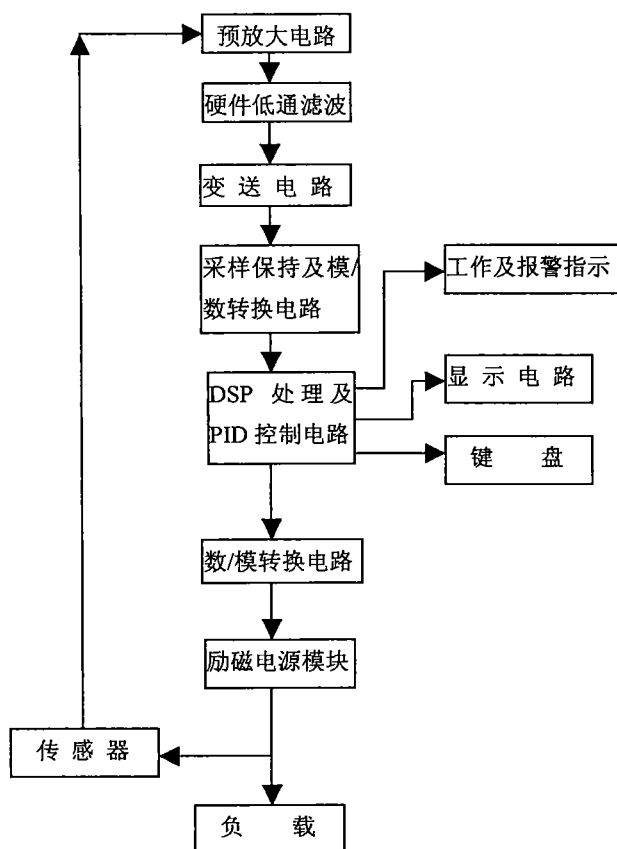


图 4.6 励磁电源控制系统硬件框图

#### 4.2.5 实际电路

图 4.7 是低通及变送电路，由三级放大电路组成，第一级放大电路对信号进行预放大，为第二级的滤波做准备，其放大倍数不能太大，以免产生失真；第二级放大电路进行低通滤波；第三级放大电路是传感器变送电路，要将其输出与第一级的输入调成线性关系，第一级来自传感器的输入信号为零时，第三级输出也为零，第一级输入为最大值时，第三级输出为 10V。

图 4.8 是 A/D 转换电路，因 AD1674 内部带有采样保持电路，所以不用在 A/D 转换电路前加采样保持电路。此电路将经过变送电路处理后通过隔离放大器送来的模拟信号进行 A/D 转换后送入 DSP 芯片。

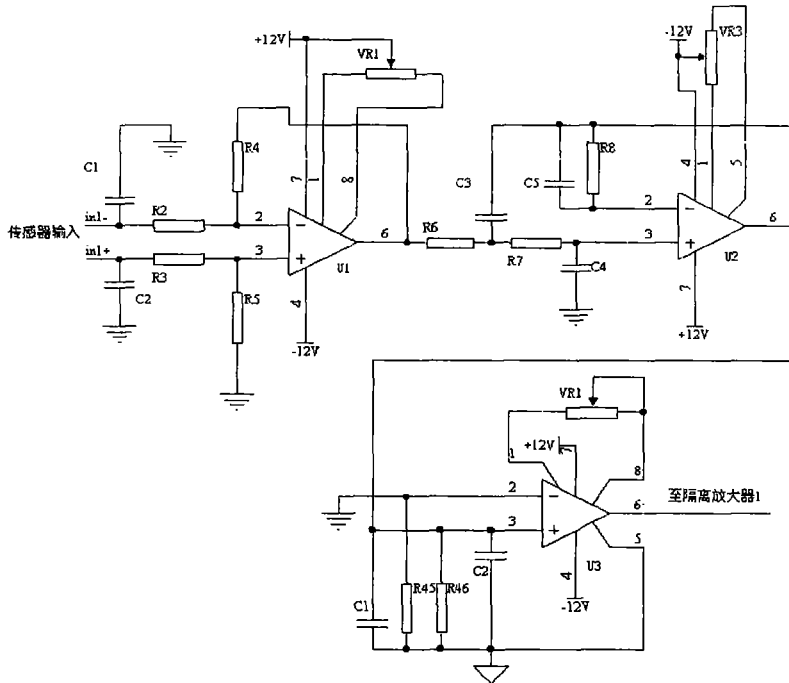


图 4.7 变送及低通滤波电路

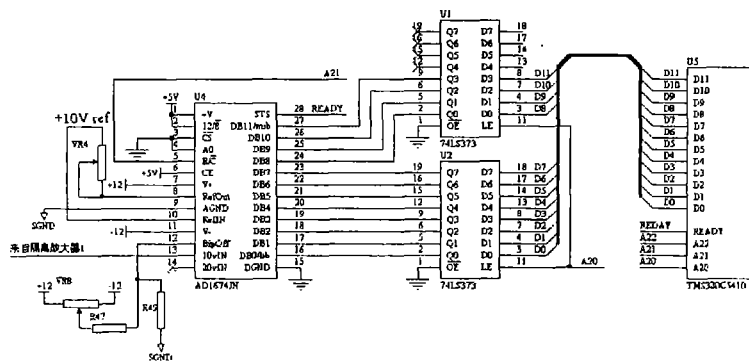


图 4.8 A/D 转换电路及其与 DSP 芯片的连接

图 4.9 为 D/A 转换电路，此电路将经过 DSP 处理后的信号送入 D/A 转换芯片，把数字量转换为模拟量去控制励磁电源模块。

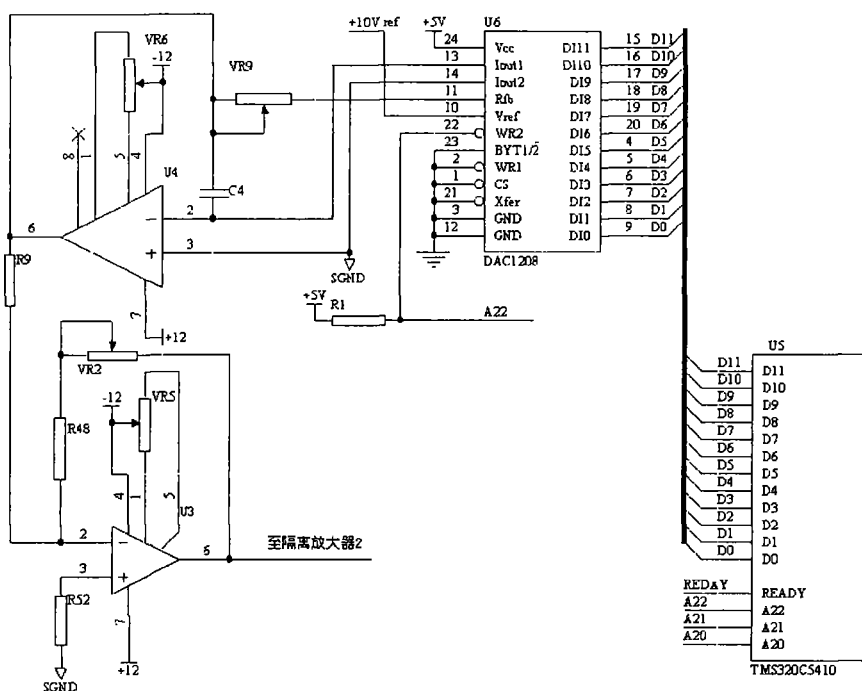


图 4.9 D/A 转换电路及其与 DSP 芯片的连接

### 4.3 系统软件设计

本软件的开发环境是德州仪器的 CCS2(5000) 软件，操作系统是微软的 Windows 98。

#### 4.3.1 软件设计思路

因基本 LMS 算法已能满足需要，所以在滤波算法的选用上，我们采用基本 LMS 算法来进行自适应滤波。A/D 转换后的数据先进行自适应 LMS 滤波，将滤波后的结果与设定的报警值相比较，如果超过了设定报警值，则向 D/A 转换器送零值，使输出为零，同时输出报警显示；如果滤波结果没有超过设定报警值，则将滤波结果进行数字 PID 调节，最后将数字 PID 运算的结果送 D/A 转换器，

由 D/A 转换器的输出电压去控制励磁电源模块。

#### 4.3.2 软件流程

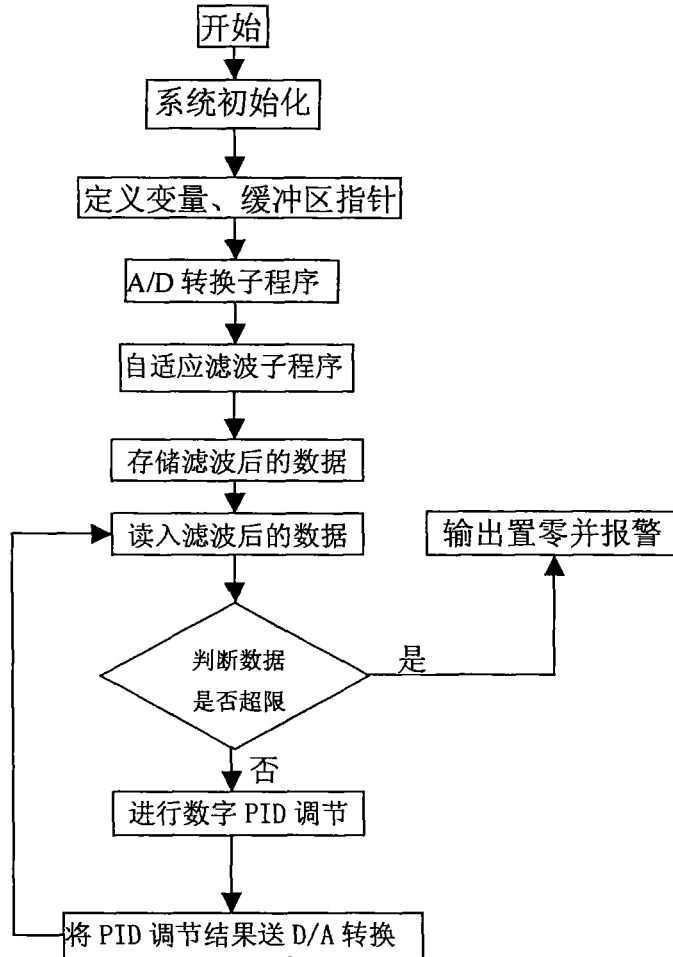


图 4.10 主程序流程图

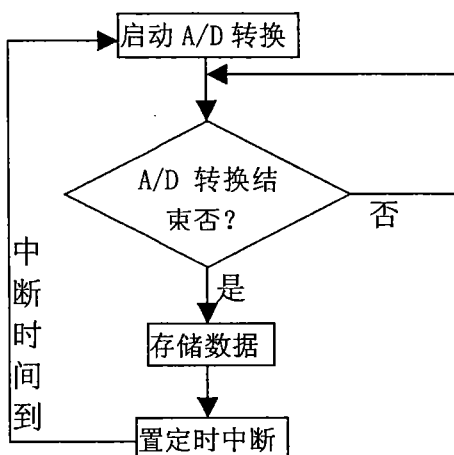


图 4.11 A/D 转换子程序

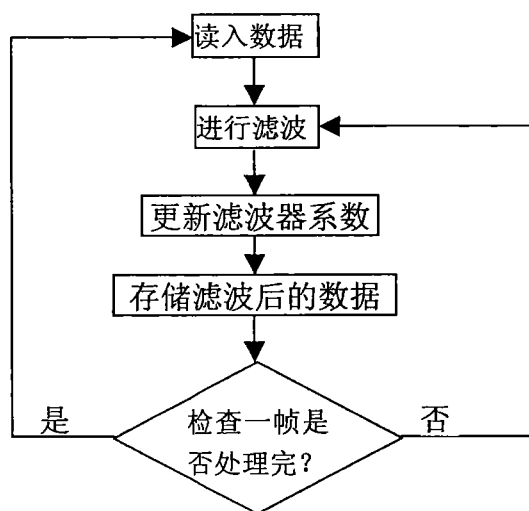


图 4.12 LMS 算法程序流程图

### 4.3.3 部分程序

由于 TMS320C5410 DSP 内部有并行的硬件乘法器、流水线结构和快速的片内存储器，它使 LMS 算法的实现变得容易且有效。

本项目用 DSP 实现比用 51 单片机来实现容易得多，所以滤波算法我们采用

横向结构的基本 LMS 算法。由于算法不复杂，为了提高运行效率，算法的实现我们用汇编语言编制。

下面给出存储器分配程序和 LMS 算法子程序。

### 1、存储器分配.cmd 文件

```
memory
{
    PAGE 0: EPROG:      origin = 0x1400,    len = 0x7c00
                VECT:      origin = 0xff80,    len = 0x80
    PAGE 1: USERREGS:   origin = 0x60,      len = 0x1c
                BIOSREGS:  origin = 0x7c,      len = 0x4
                IDATA:      origin = 0x80,      len = 0x1380
                EDATA:      origin = 0x1400,    len = 0x8000
                EDATA1:     origin = 0x9400,    len = 0x4c00
}
sections
{
    .vectors: {} > VECT PAGE 0
    .sysregs: {} > BIOSREGS PAGE 1
    .trcinit: {} > EPROG PAGE 0
    .gblinit: {} > EPROG PAGE 0
    .text:    {} > EPROG PAGE 0
    .cinit:   {} > EPROG PAGE 0
    .pinit:   {} > EPROG PAGE 0
    .sysinit: {} > EPROG PAGE 0
    .bss:     {} > IDATA PAGE 1
    .far:     {} > IDATA PAGE 1
    .const:   {} > IDATA PAGE 1
    .switch:  {} > IDATA PAGE 1
}
```

```
.sysmem:  {} > IDATA PAGE 1
.cio:      {} > IDATA PAGE 1
.stack:    {} > IDATA PAGE 1

}
```

## 2、LMS 算法子程序

本算法采用基本 LMS 算法，横向 FIR 结构，阶数为 32 阶，收敛因子  $\mu$  取 0.02，算法原理如下：

$$\textcircled{1} \quad y(n) = \sum_{i=0}^{31} w(i) * x(n-i)$$

$$\textcircled{2} \quad e(n) = d(n) - y(n)$$

$$\textcircled{3} \quad w(i) = w(i-1) + \mu * e(n) * x(n-i) \quad i=0, 1 \dots, 31.$$

```
;*****
```

```
;LMS 算法子程序:
```

```
;*****
```

```
;F_SIZE  滤波器阶数
```

```
;mu      收敛因子  $\mu$ 
```

```
;D_SIZE  数据帧长度
```

```
;*****
```

```
ADA_DP   .usect  "ada_var", 1
```

```
d_hope   .usect  "ada_var", 1    ;d(n)
```

```
d_output .usect  "ada_var", 1    ;y(n)
```

```
er        .usect  "ada_var", 1    ;e(n)
```

```
mu        .usect  "ada_var", 1    ;mu
```

```
mu_err    .usect  "ada_var", 1    ;mu*e(n)
```

```
x_new     .usect  "ada_var", 1    ;新的输入数据
```

```
a_count   .usect  "ada_var", 1    ;计数器
```

```
a_para    .usect  "bufferapara", F_SIZE ;信道参数
```

```
w_para .usect "bufferwpara", F_SIZE ;滤波系数
x_in .usect "bufferx", F_SIZE ;信道输入数据
w_in .usect "bufferw", F_SIZE ;自适应滤波输入
;初始化
```

```
ada_init:STM #w_in, AR1
          RPTZ A, #F_SIZE
          STL A, *AR1+
          STM #d_hope, AR1
          RPTZ A, #6
          STL A, *AR1+
          STM #a_para, AR1
          RPT #F_SIZE
          MVPD #w_para, *AR1+
          LD #ADA_DP
          ST #0.02, mu
          STM #1, AR0
```

;以下为 LMS 算法

```
lms_filter
          STM #F_SIZE, BK
          STM #an_para, AR2
          ADDM #1, a_count
          LD *AR6+, A
          STM #w_para, AR4
          STL A, x_new
          LD x_new, A
          STL A, *AR3+0%
          RPTZ A, #F_SIZE
          MAC *AR2+0%, AR3+0%, A
          STH A, d_hope
```



```

LD      mu_err, T
SUB     B, B
STM     #D_SIZE, BRC
RPTBD   lms_end
MPY     *AR5+0%, A
LMS     *AR4, *AR5
ST      A, *AR4
MPY     *AR5+0%, A
LMS     *AR4, *AR5
Lms_end
STH     A, *AR4
MPY     *AR5, A
MVKD    #x_new, *AR5
LMS     *AR4, *AR5+0%
STH     B, d_output
LD      d_hope, A
SUB     d_output, A
STL     A, err
LD      mu, T
MPY     err, A
STH     A, mu_err
LD      err, A
STL     A, *AR7+
LD      #D_SIZE, A
SUB     a_count, A
BC      lms_filter, AGT
RET

```

## 4.4 运行结果

图 4.13 是从传感器出来, 经过预放大电路后用示波器测得的信号; 图 4.14 是进入 DSP 芯片, 并通过基本 LMS 算法滤波后不经 PID 调节直接输出的信号; 图 4.15 是理论上没有干扰时的信号。对比图 4.14 和图 4.15, 我们可以看出滤波的效果很不错, 基本恢复出原信号。

限于开发时间关系, 本项目中没对其它 LMS 算法进行验证。

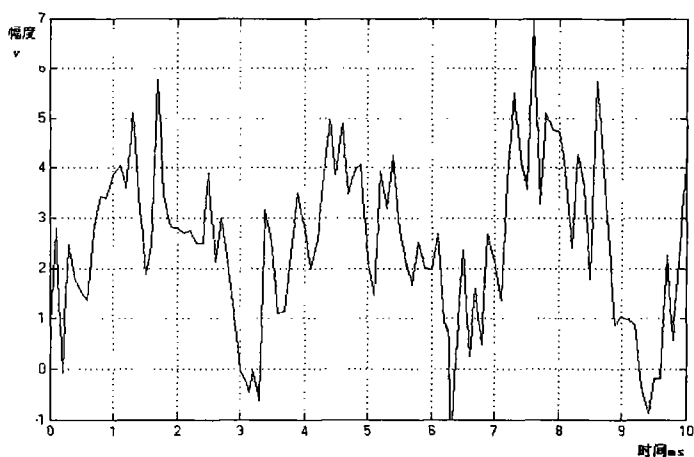


图 4.13 串入干扰的信号

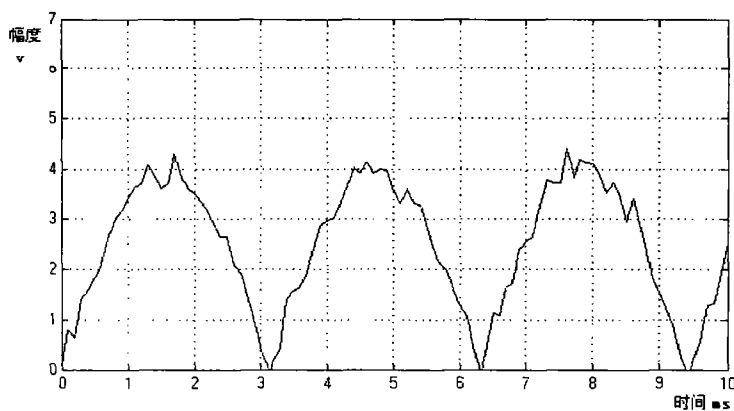


图 4.14 经过滤波后的信号

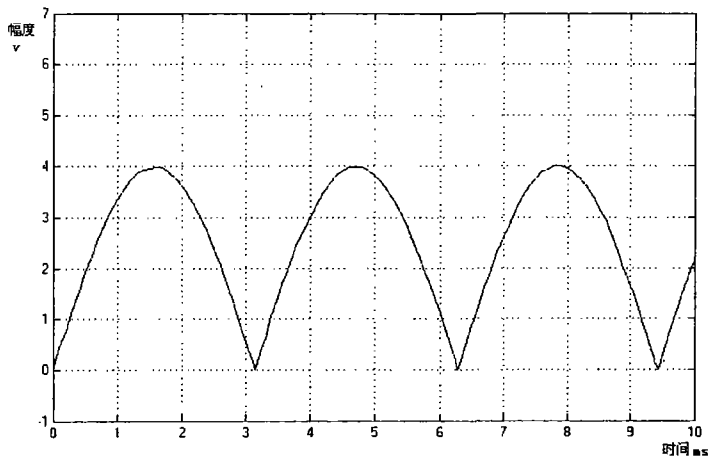


图 4.15 没有串入干扰的信号

[注]在做项目过程中，现场用示波器观看波形时因没相机而没拍下真实信号波形，图 4.13、图 4.14、图 4.15 为实验室仿真波形，真实的波形比之复杂且干扰也多得多。

## 4.5 小结

本章简要介绍了励磁电源项目中用到的几个器件的性能参数和主要特性，详细介绍了励磁电源项目软、硬件设计思路和设计过程，给出了软、硬件流程图和部分硬件电路和软件程序。充分利用 DSP 的性能优势，把 LMS 算法结合到实际项目中，取得了很好的效果。

## 结束语

实现自适应滤波的算法很多，而 LMS 算法是应用最广泛的自适应算法，同时，LMS 算法的种类也很多。本文仅对基本 LMS 算法、解相关 LMS 算法等几种算法进行了分析研究，讨论、说明了 LMS 算法的几个主要性能参数和统计、跟踪性能，还有影响 LMS 算法性能的几个因素。

通过对 LMS 算法的理论研究，本文结合实际，将理论与实践连接起来，把 DSP 应用到自动控制系统中——通过 DSP 把 LMS 算法应用于励磁电源控制系统。相对于 51 单片机系统，DSP 系统的开发难度大些，但其成本增加不大，应用效果却相当有效。

限于时间和篇幅，文中仅对基本 LMS 算法和几种 LMS 算法的原理和性能进行了阐述和简单分析，还有许多基于 LMS 算法的算法文中未提及，更多的工作留待以后去完成。

## 参考文献

- [1] 张贤达. 现代信号处理 (第二版). 清华大学出版社, 2002.
- [2] Glentis G O, Berberidis K, Theodoridis S. Efficient least squares adaptive algorithms for FIR transversal filtering. *IEEE Signal Processing Magazine* 1999, 16(4):13~41.
- [3] 刘松强. 数字信号处理系统及其应用. 清华大学出版社, 1996.
- [4] Widrow B, Stearns S D. *Adaptive Signal Processing*. New York :Prentice-Hall, 1985
- [5] 尚勇, 吴顺君, 项海格. 并行延时LMS算法. *中国科学*, 2002, 32(4):523~529.
- [6] Long G Z, Ling F Y, Proakis J G. The LMS algorithm with delayed coefficient adaptation. *IEEE Trans on Signal Processing*, 1989, 37(9):1397~1405.
- [7] Long G Z, Ling F Y, Proakis J G. Corrections to “the LMS algorithm with delayed coefficient adaptation”. *IEEE Trans on Signal Processing*, 1992, 40(1):230~232.
- [8] Herzberg H, Cohen R H, Be'ery Y. A systolic array realization of an LMS adaptive filter and the effects of delayed adaptation. *IEEE Trans on Signal Processing* 1992, 40(11):2799~2803.
- [9] Douglas S C, Zhu Q, Smith K F. A pipelined LMS adaptive FIR filter architecture without adaptation delay. *IEEE Trans on Signal Processing*, 1998, 46(3):775~779.
- [10] Matsubara K, Nishikawa K, Kiya H. Pipelined LMS adaptive filter using a new look-ahead transformation. *IEEE Trans on Circuits and Systems- II Analog and Digital Signal Processing*, 1999, 46(1):51~55.
- [11] P. S. R. .Diniz. 自适应滤波算法与实现 (第二版). 刘郁林等译. 电子工业出版社, 2002.
- [12] Texas Instruments. TMS320VC5410 Fixed-Point Digital Signal Processor Data Manual. 2000.
- [13] Andrew Bateman, Lain Paterson Stephens. DSP算法、应用与设计. 陈健等译. 机械工业出版社, 2002.
- [14] 马伟富, 雷勇, 腾欢. 自适应滤波器 (LMS) 算法及其在DSP上的实现. *四川大学学报 (自然科学版)*. 2004, VOL 41:470~473.
- [15] 王金龙, 沈良等. 无线通信系统的DSP实现. 人民邮电出版社, 2002.

- [16] 赵红怡, 张常年. 数字信号处理及其MATLAB实现. 化学工业出版社, 2002.
- [17] 李刚. 数字信号微处理器的原理及其开发应用. 天津大学出版社, 2000.
- [18] Magno T. M, da Silva, Max Gerken, Maria D. Miranda. An Accelerated Constant Modulus Algorithm. ITS2002, Natal, Brazil.
- [19] Vitor H. Nascimento. Analysis of the Hierarchical LMS algorithm. IEEE Signal Processing Letters, 2003, 10(3):78~81.
- [20] Michael Reuter, James R. Zeidler. Nonlinear Effects in LMS Adaptive Equalizers. IEEE Transactions on Signal Processing, 1999, 47(6):1570~1579.
- [21] Widrow, S D Stears. Adaptive Signal Processing. Englewood Cliffs. Prentice Hall, 1985
- [22] 胡细勇, 朱权, 伍继雄. 基于 DSP 技术和自适应 LMS 算法的无线同频中继站的探讨. 移动通信, 2000. 5. <http://www.mc21st.com/magazine/2000-5/y05.htm>
- [23] 周国强, 黄本雄, 吕谦. IP 电话回声消除器的设计及其DSP实现. 电子工程师, 2003, 1(3). <http://www.bjx.com.cn/files/wx/dzgcs/2003-3/8.htm>
- [24] V. J. Mathews. performance analysis of adaptive filters equipped with dual sign algorithm. IEEE Transactions on Signal Processing, 1991, 39(1):85~91.
- [25] A. V. Oppenheim, R. W. Schaffer. 数字信号处理. 北京: 科学出版社, 1983.
- [26] 雷勇, 马伟富, 涂国强. 基于 DSP 的微动机器人位置控制的实现. 电子技术应用, 2005, 1:62~64.
- [27] 王锦标. 计算机控制系统. 清华大学出版社, 2004.
- [28] 李哲英, 骆丽, 刘元盛. DSP 基础理论与应用技术. 北京航空航天大学出版社, 2002.
- [29] Yuantao Gu; Kun Tang; Huijuan Cui. LMS algorithm with gradient descent filter length. IEEE Signal Processing Letters. 2004, 11(3):305~307.
- [30] 胡广书. 数字信号处理-理论、算法与实现(第二版). 清华大学出版社, 2003.
- [31] 邹理和. 数字滤波器. 北京: 国防工业出版社, 1982.

## 本文作者在读期间科研成果简介

### 学术论文

- 1、雷勇，马伟富，涂国强. 基于 DSP 的微动机器人位置控制的实现。电子技术应用，2005, 1:62~64。
- 2、马伟富，雷勇，腾欢. 自适应滤波器（LMS）算法及其在 DSP 上的实现。四川大学学报(自然科学版), 2004, VOL 41:470~473。
- 3、雷勇，翁嫣琥，马伟富. 虚拟仪器技术在智能测控系统中的应用。四川大学学报(自然科学版), VOL 41:461~465。

### 工程实践

- 1、1:1 摩擦试验台智能测控系统的研制，2003. 7-2004. 6
- 2、机车用空调电源的研制，2004. 3-2004. 9
- 3、励磁电源控制系统的研制，2004. 10-2004. 12

## 声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得四川大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

本学位论文成果是本人在四川大学读书期间在导师指导下取得的，论文成果归四川大学所有，特此声明。

研究生 马伟富

指导教师 雷勇

2005 年 5 月 25 日

2005 年 5 月 26 日



## 致谢

衷心感谢各位评审专家、教授对本论文提出宝贵的意见。

本文从选题、收集资料、软、硬件调试直至完成，都得到了导师雷勇副教授的悉心指导和帮助。

在三年的研究生学习和生活中，雷老师给予了我无微不至的关怀和谆谆教诲，让我受益良多。雷老师高度的责任心和严谨的治学态度都将对我今后的工作和学习产生深远的影响。在此特向恩师表示由衷的敬意和感谢。

同时，我向在我学习和设计过程中给了我很大帮助的涂国强老师、李雷老师及覃琴师弟、刘玺师弟、韦伟师弟、翁嫣琥师妹表示诚挚的谢意。

此外，我还要感谢我的妻子张良萍多年来对我始终如一的支持、关心，她的支持和关心是我完成学业最大的动力。