

密级:_____



中国科学院大学
University of Chinese Academy of Sciences

硕士学位论文

水声通信中 Turbo 均衡技术的研究

作者姓名:_____唐怀东_____

指导教师:_____朱敏 研究员 中国科学院声学研究所_____

_____武岩波 副研究员 中国科学院声学研究所_____

学位类别:_____工学硕士_____

学科专业:_____信号与信息处理_____

研究所:_____中国科学院声学研究所_____

2013 年 5 月

Research of Turbo Equalization for Underwater
Acoustic Communication System

By
Tang Huaidong

A Dissertation Submitted to
The University of Chinese Academy of Sciences
In partial fulfillment of the requirement
For the degree of
Master of Signal and Information Processing

Institute of Acoustics, Chinese Academy of Sciences

May, 2013

中国科学院声学研究所

学位论文原创性声明和使用授权说明

原创性声明

本人郑重声明:本论文的所有工作,是本人在导师的指导下,独立进行研究工作所取得的成果。除文中已经注明引用的内容外,本论文不含任何其他个人或集体已经发表或撰写过的作品或成果。对本文的研究做出重要贡献的个人和集体,均已在文中以明确方式标明。本声明的法律结果由本人承担。

论文作者签名:

日期: 年 月 日

学位论文使用授权说明

本人完全了解中国科学院研究生院关于收集、保存、使用学位论文的规定,即:

- 按照中国科学院研究生院要求提交学位论文的印刷本和电子版本;
- 中国科学院研究生院与中国科学院声学研究所有权保留学位论文的印刷本和电子版,并提供目录检索与阅览服务;
- 中国科学院研究生院与中国科学院声学研究所可以采用影印、缩印、数字化或其它复制手段保存论文;

(保密论文在解密后遵守此规定)

论文作者签名:

导师签名:

日期: 年 月 日

摘 要

在低信噪比的水声通信系统中,采取卷积码和 RS 码来提高信息传送的可靠性。对于卷积码,采用序贯译码方式,来简化译码算法,提高译码效率。而费诺算法是序贯译码方式中空间开支小,运算量小,实现比较简单,现实应用比较广泛的一种算法。而对于 RS 码,采用 BM 算法进行译码。

通过 matlab 仿真,确定译码参数,并在 PC/104 上运行,测试效率。

通过测试得到的数据,可以得到几个结论。卷积码序贯译码在约束长度为 24 的情况下,误码率可以很好的满足要求,与书上的理论相一致,而运行时间有一定的延迟,因此系统需要缓冲器来存储译码数据;而 $RS(15, 9, 7)$ 采用 BM 译码算法,误码率有很大的改善,而且运行时间很短,因此系统不需要缓冲器来存储译码数据。

关键词

低信噪比 水声通信 卷积码 RS 码

ABSTRACT

In the low signal to noise ratio of acoustic communication system, convolutional code and RS code are adopted to improve the reliability of information transmission. For convolutional code, using sequential decoding method to simplify the decoding algorithm and to improve the efficiency. The fano algorithm is one of the sequential decoding methods which is small space spending, relatively simple to achieve and used widely in reality. So, finally taking it as the decoding algorithm. For RS code, BM algorithm is almost the most widely one among all decoding algorithms.

Determining the coding parameters by matlab simulation and test decoding efficiency in PC/104.

Through test data, several conclusions can be obtained. With sequential decoding algorithm, the error of convolutional code which constraint length is 24 can be very good to meet the requirements and be consistent with the theory of the book, also the running time is with delay. Therefore, the system needs to a buffer to store the decoding data; The $RS(15, 9, 7)$ with BM decoding algorithm, the error rate has been greatly improved and the running time is very short. So the system does not need to buffer the decoding data.

Keywords

Low SNR Underwater acoustic communication Convolutional code RS code

目录

摘 要	I
ABSTRACT	III
目录	i
 第 一 章 引 言	 1
1.1 论文研究的背景与意义	1
1.2 水声通信技术发展概况	2
1.3 水声通信中的均衡技术	3
1.3.1 水声通信中使用均衡器的必要性	4
1.3.2 均衡技术的发展	5
1.3.3 Turbo 均衡技术的发展	6
1.4 论文的研究内容及章节安排	7
 第 二 章 水声信道特点与自适应均衡技术	 10
2.1 水声信道的特点	10
2.1.1 水声信道模型	10
2.1.2 双扩散特征	11
2.1.3 有限带宽	12
2.1.4 声波传播损失	13
2.1.5 环境噪声	14
2.2 自适应均衡技术	15
2.2.1 自适应均衡技术综述	15
2.2.2 自适应均衡算法	16
2.3 Turbo 均衡	23
2.3.1 Turbo 均衡原理	24
2.3.2 Turbo 均衡算法	25
2.4 本章小节	27

第三章 基于先验信息 MMSE 准则的线性 Turbo 均衡算法	29
3.1 MMSE 均衡算法	29
3.1.1 MMSE 估计器	30
3.1.2 映射	33
3.1.3 算法优化与总结	35
3.2 低复杂度近似算法	39
3.3 仿真与分析	40
3.3.1 不同均衡算法之间的比较	40
3.3.2 不同调制方式之间的比较	42
3.3.3 不同编码方式之间的比较	43
3.3.4 不同删余方式之间的比较	44
3.3.5 不同交织长度之间的比较	46
3.4 分数间隔线性 SISO 均衡算法	47
3.5 Turbo 均衡与 Turbo 联合迭代	50
3.6 采用预编码的线性 SISO 均衡算法	50
3.7 本章小结	51
第四章 软迭代信道估计算法	53
4.1 RS 码与水声通信	53
4.2 RS 码概述	53
4.3 RS 码的编码	54
4.3.1 基于乘法形式的 RS 编码器	54
4.3.2 基于除法形式的 RS 编码器	55
4.4 RS 码的译码	57
4.5 Berlekamp-Massey 硬判决算法	58
4.5.1 Berlekamp-Massey 迭代原理 ^[2]	58
4.5.2 BM 算法的主要步骤及实现 ^[2]	62
4.5.3 钱搜索与 Forney 算法	63

4.5.4 BM 算法性能仿真	64
4.6 本章小结	65
第 五 章 EXIT 图在 Turbo 均衡中的应用	67
5.1 PC/104 简介	67
5.2 卷积码序贯译码在 PC/104 上运行	68
5.3 RS 码 BM 算法译码在 PC/104 上运行	69
5.4 本章小结	71
第 六 章 水声通信系统 Turbo 均衡算法测试数据分析	73
6.1 研究工作总结	73
6.2 下一步工作	73
6.2.1 费诺算法的进一步改进	73
6.2.2 Turbo 码等其他码字的研究	74
参考文献	76
个人简历及论文发表	84
致谢	86

第一章 引言

1.1 论文研究的背景与意义

海洋工程技术、宇航空间技术和核能科学技术并列作为当代技术革命中三大尖端技术。随着世界经济的飞速发展和人口的不断增加,人类今天正面临着人口、资源和环境三大难题。随着消耗,陆地上的资源正在不断减少,为了生存和发展,人们早已开始向海洋寻找资源供给。

海洋占地球表面积的 71%,它拥有 14 亿立方千米的体积。在海底及海洋中,蕴藏着极其丰富的生物资源及矿产资源。仅大洋锰结核的储量就有约 3 万亿吨,其中锰 6233 亿吨、铜 110 亿吨、钴 94.5 亿吨、镍 233 亿吨、铁 4300 亿吨、铝 883 亿吨等。与陆地上已经探明的储量相比,钴是其 5250 倍,铁为 4.3 倍,其它都在 33 倍以上。海洋还是一个无比巨大的能源库,全世界海洋中储存着约 1350 亿吨石油,约 140 万亿立方米的天然气^[1]。因此,大洋海底的探索 and 开发具有极强的吸引力。

在过去的几十年中,随着人类海洋开发、海洋利用和海洋探索活动的日益增加,人类对水下数据获取和数据传输技术的需求也越来越大。水下的数据由传感器传送到海洋表面,从那里可以将数据经由卫星传发给远处的数据处理中心。

除极低频率外,电磁波在水中的衰减很快,因此不能长距离传播。穿透力较强的长波能力也极为有限,同时还需要体积庞大的天线和大功率发射机来支持,传输速率也非常低^[2]。此外,中微子和蓝绿激光也可以用于水下通信^[3],但是中微子目前还不能方便、可靠的得到,蓝绿激光的传播距离有限^[4]。声波可以在水中传播很远的距离,是海洋中的主要信息载体。水下声波是目前最可行的水下探测和通信手段。海洋声学技术在海洋中用于探测、通信、导航和定位等领域,近年来随着海上军事活动及海洋开发的迅速增加,海洋声学技术获得了迅速的发展^[5,6]。

水声通信是一门综合学科,数字信号处理、无线电通信、移动通信、卫星通信,扩频通信以及软件无线电技术和声纳技术的成果都可以借鉴,计算机技术、微电子技术以及高速数字信号处理器 (DSP) 技术的发展也为水声通信的不断进步提供了坚实的硬件支持。因此,水声通信技术本身可以通过吸收借鉴以及综合各相关学科的优秀成果得到发展。同时水声技术又有着自己的特色,在水声通信发展

的初期,由于要求传递的信息量往往不是很大,人们对于水声通信系统的性能要求通常并不是很高。随着人类海洋研究活动的不断深入,对于水声通信系统的效率和可靠性也提出了更高地要求,这使得仅使用传统的通信技术已经不能达到相应的技术要求。因此,许多新方法、新技术引入到了水声通信领域^[7--14]。

近年来,在水声通信领域中均衡技术应用得越来越广泛。均衡技术最初应用于无线电通信领域,尤其是电话系统中。之后,随着计算机和网络的发展和普及,在调制解调器中也广泛采用了均衡技术。由于水声通信系统工作于复杂的湖泊、海洋等水声信道之中,因此,在水声通信系统中采用均衡技术消除或减弱信道对信号传输的负面影响是提高系统有效性和可靠性的必要途径。从另一角度看,也正是由于水声通信信道的多样性以及复杂性进一步促进了均衡技术的快速发展。因而,对于水声通信系统中均衡器的应用及其实现的研究已成为近年来水声通信领域的另一研究热点。

1.2 水声通信技术发展概况

人类关于水下发送、接收信息的想法可以追溯到几百年前。早在 1490 年,意大利的达·芬奇在他的摘记中就有“将长管的一段插入水中,将管的开口放在耳旁就听到远处的航船”的记载。而最具现代意义的真正的水声通信出现在第二次世界大战中,主要用于军事上。例如 45 年美国开发的用于潜艇间通信的水声电话,采用单边带¹调制技术。水声电话技术比较成熟,目前仍广泛应用于潜艇间以及潜水员与母船间的通信。但其明显的缺点是功能单一,只能传输语音。

由于客观的需求增加,从二十世纪七十年代开始,军事领域和民用领域都对水声通信技术产生了大量的需求,如在军事领域中,舰艇间的通信,对水下航行器实施监测和导航,水雷的远程声遥控等使得水下通信技术的研究得到人们的高度重视,水声通信技术的重要性也日益突出;在民用领域,渔业资源的开发利用,海上钻井平台和船只的应急维护,水下机器人的研制,水下资源勘探等的发展,对水声通信技术也提出了新的要求。这使得水声通信进入了一个发展相对迅速的阶段。

在水声通信技术快速发展的同时,其它领域的技术,尤其是电信、电子和计算

¹SSB,single sideban

机技术以更为迅猛的速度日新月异地前进。这极大地促进和支持了水声通信技术的发展,水声通信技术发生了深刻地变化,水声通信的面貌焕然一新。

首先发展起来的数字式水声通信系统采用的是非相干水声通信技术。非相干水声通信技术采用多进制频移键控信号(MFSK)加编码的技术克服多径引起的干扰,其带宽利用率较低,传输速率一般为数百 bits/s,但其具有好的鲁棒性,因此得到了广泛应用。相干水声通信技术采用多进制相移键控信号(MPSK)、空间分集、自适应均衡器、编码和多普勒补偿等技术,带宽利用率比非相干技术提高了一个数量级,一般传输速率为数千至上万 bits/s。相干水声通信技术仍处于迅速发展和完善之中,是当前水声通信技术的发展主流。由于水声信道具有时变-空变特性,任何单一的通信技术都很难满足水声通信系统高可靠传输数据的要求。因此,在一个水声通信系统中往往是数种通信技术的联合,共同发挥效用,以期达到优良的系统通信性能。在水声通信领域,空间分集技术与自适应均衡技术的结合逐渐成为研究热点。采用这种方式可以同时利用均衡以及分集处理在时间域和空间域获益,从而更为有效地消除水声信道的多径效应。目前水声通信技术仍在不断发展,各国都在研究把各种新技术应用到水声通信中,如包括采用正交频分复用调制(OFDM)的多载波通信技术、采用码分多址(CDMA)的水下组网技术等^[15--17]。我国是在八十年代中期开展水声通信技术研究的,中科院声学所、厦门大学、哈尔滨工程大学等都在开展高速水声通信研究,其中中科院声学所在“七五”期间研制了频移键控水声通信样机,在“八五”、“九五”期间开展了相移键控水声通信机的研究^[17],并被列为国家“863”计划智能机器人主题的预研课题。在“十五”期间声学所承担了某潜水器声学系统的研制任务,在已有技术基础上进一步发展完善,研制一套实用的中程高速水声通信机。厦门大学在厦门湾进行了相干通信的试验^[9]。哈尔滨工程大学在松花湖进行了相干通信和 OFDM 通信的试验。

1.3 水声通信中的均衡技术

水声信道是一多径、色散、时变和深度衰落的信道,声波在其中的传播行为十分复杂。由于水声信道十分恶劣,信道的噪声干扰、时变特性、严重的多径效应和复杂多变的传播环境使水声通信系统中存在着十分严重的码间干扰(ISI)和噪声;界面和介质的起伏也导致了信号的相位变化较剧烈。均衡技术可以补偿信道多

径效应和多普勒扩散引起的信道畸变从而减少码间干扰;分集技术则用来补偿信道衰落损耗,而信道编码是通过在发射信息中加入冗余数据来改善数据的纠错性能,从而提高通信系统的接收性能。本文研究了水声相干通信系统的 Turbo 均衡技术,与信道编码相联合迭代,以达到提高通信系统性能的目的。

1.3.1 水声通信中使用均衡器的必要性

在水声通信系统中,信道的多径效应会产生码间干扰,使发射信号发生畸变,从而在接收端产生误码。

设任一基带信号为:

$$s(t) = \sum_{n=0}^{\infty} a_n g(t - nT) \quad (1-1)$$

其中, $g(t)$ 是要选择的基本脉冲形状,用于控制传输信号的频谱特性。 a_n 是由 M 个点组成的信号星座图中选取的传输信息符号的序列, T 是符号区间 ($1/T$ 就是符号率)。

该基带信号经过频率响应为 $C(f)$ 的信道后,接收的信号可以表示为:

$$r(t) = \sum_{n=0}^{\infty} a_n h(t - nT) + n(t) \quad (1-2)$$

其中, $h(t) = g(t) * c(t)$, $c(t)$ 是信道冲击响应, $n(t)$ 代表信中加入的高斯白噪声。为了表示码间干扰,假设接收信号通过一个接收滤波器,然后以每秒 $1/T$ 的采样率进行采样。一般来说,在接收端最佳滤波器是与接收信号脉冲 $h(t)$ 相匹配的匹配滤波器,所以这个滤波器的频率响应为 $H^*(f)$ 。滤波器的输出可以表示为:

$$y(t) = \sum_{n=0}^{\infty} a_n x(t - nT) + v(t) \quad (1-3)$$

其中, $x(t)$ 是接收滤波器的信号脉冲响应,即 $X(f) = H(f)H^*(f)$, $v(t)$ 是接收滤波器对噪声 $n(t)$ 的响应。如果对 $y(t)$ 在时刻 $t = kT, k = 0, 1, 2, \dots$ 进行采样,则有:

$$y(kT) = \sum_{n=0}^{\infty} a_n x(kT - nT) + v(kT) \quad (1-4)$$

记为:

$$y_k = \sum_{n=0}^{\infty} a_n x_{k-n} + v_k, \quad k = 0, 1, 2, \dots \quad (1-5)$$

样本值 y_k 可以表示为:

$$y_k = x_0 \left(a_k + \frac{1}{x_0} \sum_{\substack{n=0 \\ n \neq k}}^{\infty} a_n x_{k-n} \right) + v_k, \quad k = 0, 1, 2, \dots \quad (1-6)$$

x_0 是任意加权因子,为了方便,将其置为 1,那么式 (1-6) 可改写为:

$$y_k = a_k + \sum_{n=0}^{\infty} a_n x_{k-n} + v_k, \quad k = 0, 1, 2, \dots \quad (1-7)$$

其中, a_k 代表了在第 k 个采样时刻所期望的符号,而 $\sum_{\substack{n=0 \\ n \neq k}}^{\infty} a_n x_{k-n}$ 项代表了码间干扰。

由 (1-7) 可见,水声通信系统为了获得高可靠地数据通信,必须克服由信道产生的码间干扰,而均衡技术是克服信道多径效应、减少码间干扰的有效手段。因此,在水声通信系统中使用均衡器是非常必要的。

1.3.2 均衡技术的发展

均衡技术最早应用于无线电通信领域,主要用于消除由信道响应引起的码间干扰。均衡技术大致分为两大类:线性均衡及非线性均衡器。常用的最大似然序列估计器 (MLSE)、判决反馈均衡器 (DFE) 和 Turbo 均衡器都属于非线性均衡器。基于最小均方 (LMS) 算法的线性均衡器算法成熟且简单,但是由于其适用于多种码间干扰不很严重的场合,而对于复杂的水声信道并不适用。非线性均衡器中的最大似然序列估计器由于计算量过大,实际中也难以应用。目前,水声通信系统中应用最广泛的是判决反馈均衡器 (DFE)。

由于水声信道是时延和多普勒双扩散信道,上世纪九十年代,在自适应判决反馈均衡器中加入了信号相位补偿器,使均衡器的性能有了长足的进步。

从现代水声通信技术的发展过程来看,自适应均衡技术在其中扮演着重要的角色。利用自适应均衡技术来提高水声通信系统的传输速率和频带利用率,已经成为现代水声通信系统的重要特征。

由于水声信道具有时变-空变及衰落特性,因此,单一的均衡技术已经不能满足现代水声通信系统的要求。如今,自适应均衡技术已经发展到了各种类型的均衡器互相借鉴、融合,以及与通信系统中的其他环节相互联合的阶段。例如,将判决反馈均衡器与最大似然序列估计器相结合^[18]、联合判决反馈均衡器与线性预

测方法跟踪衰落信道特性^[19]、最大似然序列估计器与 FIR 滤波器联合工作^[20],以及分集技术相结合的方法^[21]等等。而另外一种常见的方式则是自适应均衡方法联合各种编码以及调制技术,例如:极大似然序列估计器与编码调制技术相结合^{[22][23]},判决反馈均衡器结合编码技术以及编码调制技术^[24--26],Turbo 均衡器与译码器联合迭代^[27--39]等新方法,新理论。而其中的 Turbo 均衡技术是最近几年研究的比较广泛的一种均衡技术,该均衡技术能够通过迭代有效的利用译码器反馈的软信息,从而提高均衡性能。下面对这个均衡算法作简要说明。

1.3.3 Turbo 均衡技术的发展

Turbo 码由法国的 C.Berrou^[40] 等提出,已经被广泛用于无线和水声通信系统中。Turbo 码具有反馈迭代的译码结构和对软信息的利用能力,因此具有极为优越的性能,也能够与其它技术相结合。

利用了 Turbo 码的译码原理,在接收端,均衡器与译码器的联合处理,即一种迭代算法在对同一组接收到的数据进行重复地均衡和译码,这种处理就是 Turbo 均衡^[37,41]。在 Turbo 均衡算法中,性能最好的是最大后验概率 (MAP) 算法 [],但是其计算复杂度极高,难以在实际应用中使用。因此,人们更倾向于寻找一些降低复杂度而性能次优的算法。Ariyavisitakul 和 Li^[42] 提出了一种联合编码和均衡的方案,和以往的接收机不同,这种接收机采用了卷积码和一个判决反馈均衡器 (DFE),在 DFE 中,DFE 的前向滤波器中输出的软信息作为维特比译码器的输入,而维特比译码器输出的硬判决信息作为 DFE 中反馈滤波器的输入,从而形成一个回路,通过不断的迭代来提高均衡和译码的性能。相比于上述算法中 DFE 利用维特比译码器^[43] 输出的硬判决信息,Marandian 等人^[44] 于 2001 年提出一种利用译码器软信息的基于 DFE 的 Turbo 均衡算法。该算法能够避免硬判决导致的信息损失,从而更有效的提高均衡性能。Wang 和 Proor^[45] 提出一种类似 Turbo 均衡的系统,作为 CDMA 多用户检测器的一部分。其中基于 Turbo 均衡的迭代结构采用了一个线性均衡器 (LE) 和 MAP 译码器来减少码间干扰。用 LE 均衡器代替 MAP 均衡器大大减少了计算量。而 Glavieux^[46] 等人提出了用基于线性滤波器的软干扰抵消器 (SIC) 代替 MAP 均衡器,以达到减少计算量的目的。这个 SIC 的系数由基于最小均方误差 (LMS) 的更新算法给出。后来有人对这种方法进行了改

进,为了使 LMS 算法的结果趋近于 MAP 均衡的结果,在各种比特信噪比 (SNR) 下,将 MAP 均衡的结果进行线性估计,然后存入表中供接收机使用^[47]。^[48] 的近似方法类似于^[46] 中的方法,只是该近似方法适用于已知部分信道响应的磁记录应用中。均衡滤波器的输出被认为是可靠地度量,接收机通过该度量来决定是否使用线性均衡算法代替 MAP 算法。另外一种降低 MAP 均衡器复杂度的通用方法是减少状态转移图中的状态个数,具体参考^[49]。本文提到的这些近似算法都存在一个巨大缺陷²,那就是均衡器的运算复杂度随着信道冲击响应的长度以及符号映射的大小呈指数型增长。Tüchler^[32,33,50] 等人提出基于先验信息 MMSE 准则的线性 Turbo 均衡算法。与传统 MMSE 均衡算法相比,此时的符号分布已经不是独立同分布的了,因此该算法不仅考虑噪声的分布情况同时也考虑了符号的分布。通过利用此中的软信息可以很大程度地提高均衡性能。

目前为止,Turbo 均衡性能的好坏,还不能从理论上加以解释,而是从实验中证明其优异的性能。对 Turbo 均衡算法进行研究,不仅要考虑到其性能的好坏,也要考虑到其算法的复杂度以及计算量的问题。同时也要考虑其在实际工程中能否适用。

1.4 论文的研究内容及章节安排

本论文在分析了基于先验信息 MMSE 准则的线性 Turbo 均衡算法的基础上,将 Turbo 编码与该均衡算法相结合,进行理论研究及仿真分析。为了应用于水声通信系统,针对水声信道的特点,提出一种软迭代信道估计算法,并与上述的 Turbo 均衡方案联合,最终用于水声相干通信中。

论文由七章组成,各章内容安排如下:

- **第一章 引言** 首先介绍了水声通信的发展概况,接着结合水声信道的特征,说明水声通信系统中使用均衡技术的必要性,以及均衡技术的研究及发展情况。
- **第二章 水声信道特点与自适应均衡技术** 介绍水声信道的特点以及当前水声相干通信中的自适应均衡技术。

²这个缺陷在经典的 Turbo 均衡方案中就存在

- **第三章 基于先验信息 MMSE 准则的线性 Turbo 均衡算法** 介绍 Turbo 均衡的系统模型及原理,并与传统均衡方案的比较,且详细介绍基于先验信息 MMSE 准则的线性 Turbo 均衡算法并仿真分析其性能。
- **第四章 软迭代信道估计算法** 介绍了一种针对水声信道特点的软迭代信道估计算法,并仿真分析其性能。
- **第五章 EXIT 图在 Turbo 均衡中的应用** 介绍一种外部信息转移图 (EXIT),该图能够预测随着迭代次数的增加,自适应 Turbo 均衡的性能,对于设计 Turbo 均衡算法有很大的帮助。
- **第六章 水声通信系统 Turbo 均衡算法湖试数据分析** 介绍湖试试验的概括,对采集的数据后处理并对结果进行分析。
- **第七章 结论与展望** 介绍论文的研究结论和未来的研究工作。

第二章 水声信道特点与自适应均衡技术

2.1 水声信道的特点

根据通信系统的使用环境,通信信道可分为:无限带宽信道,限带信道和多径衰落信道^[2]。这三类信道均受到噪声的干扰。无限带宽信道具有固定的信道衰减和相移,限带信道具有确定的或随时间缓慢变化的信道冲击响应,多径衰落信道具有随机时变冲击响应特性。水声信道的传播速度低,可用带宽窄,是一个时变、多径衰落信道。温度、盐度、深度、风、浪、流以及航运、海底工程等人类活动都对水声信道产生直接而显著的影响,与无线电波信道相比,水声信道要恶劣得多。深刻了解水声信道的传播特点,以及这些特点对水声信号产生了什么影响,是研究水下声通信技术的主要切入点之一。

2.1.1 水声信道模型

对于水声信道这样的物理信道,会导致发送信号的时变、多径传播,在数学上我们可以将其表征为时变线性滤波器。该滤波器可以表征为时变信道冲击响应 $c(\tau, t)$,它是信道在 $t - \tau$ 时刻加入冲击而在 t 时刻的响应。对于输入信号 $s(t)$,信道输出为:

$$\begin{aligned} r(t) &= s(t) * c(\tau, t) + n(t) \\ &= \int_{-\infty}^{\infty} c(\tau, t) s(t - \tau) d\tau + n(t) \end{aligned} \quad (2-1)$$

用来表征通过物理信道的多径信道传播的模型是式 ((2-1)) 的一个特例,该特例中的时变冲击响应为:

$$c(\tau, t) = \sum_{k=1}^L a_k(t) \delta(\tau - \tau_k) \quad (2-2)$$

式中, a_k 表示 L 条多径传播路径上可能的时变衰落因子, τ_k 表示相应地时间延迟。

将式 (2-2) 代入式 (2-1) 中,则接收信号可以表示为:

$$r(t) = \sum_{k=1}^L a_k(t) s(t - \tau_k) + n(t) \quad (2-3)$$

因此,接收信号由 L 个路径分量组成,其中每一个分量的衰减因子为 a_k ,延迟为 τ_k 。

水声信道是一种典型的时延-多普勒双扩散信道,因此,简单地将水声信道表示成抽头延迟线模型是不合适的。在第四章介绍信道估计算法时将给出水声信道的双扩散模型。

2.1.2 双扩散特征

多径效应

在水声通信系统中,水声信道的多径效应被认为是水声通信中遇到的最大困难。由于海水介质的非均匀性、声传播信道中的海底海面反射、海洋中的各种各样的反射体和散射体的存在以及声波在水声较慢的传播速度,水声信道必然存在多径现象,也就是说在一定波束宽度内发出的声波可沿几种不同的路径到达接收点。由于声波在不同路径中传播时路径长度的差异,到达接收点的声波能量和时间也不相同,从而引起信号的衰落及波形的畸变。

多径效应的形成于海洋环境和信号频率有关,其形成主要机理是声线弯曲和海底、海面的反射,海曙中内部结构如内波、紊流、潮汐等的影响,以及声源和接收机平台的运动等。当声波在不同的层、海底和海面间传播会造成多次的反射和折射,从而形成各个不同的传播路径。水下声信道在相干时间长度内,可简化为相干多径信道,仅仅存在多径效应。

多径传播时影响水声通信系统性能的重要因素之一。多径传播引起信号的时延扩展。在频域上表现为信号的频率选择性衰落。时延扩展又被称为多径扩展,会引起信号的码间干扰。由于海水中内部结构(如内波、水团、湍流等)的影响,多径结构通常是时变的,多径的时变性引起传输信号的频率扩散,导致传输信号产生时间选择性衰落,实际中这两种衰落经常伴随在一起的。

在数字通信系统中,多径传播会造成码间干扰。在无线电信道中,码间干扰通常为几个码元宽度,而在水平传播的水声信道中,对中、高数据率的浅海信道码间干扰将由几十和几百个码元宽度。多径传播造成的码间干扰是影响水声信道系数数据率的主要因素。目前,人们常用均衡等技术来对抗多径引起的码间干扰。

多普勒频移

由于发射机和接收机之间的相对运动以及海水流动和湍流的作用,声波在浅海信道传播过程中会产生一定的频率漂移,即为多普勒频移。当收、发机之间有相对运动的情况时,假定发射端和接收端的运动速度分别为 ν_s 和 ν_r ,信号的发送频率为 f_s ,则当信号到达接收端时,接收信号的频率 f_r 可以表示为:

$$f_r = f_s \frac{c - \nu_r}{c - \nu_s} \quad (2-4)$$

上式中, ν_s 和 ν_r 分别是发射机和接收机的速度,单位为米/秒 (m/s), c 为海水中的声速,一般理论值为 1500m/s, f_s 和 f_r 的单位是赫兹 (Hz)。

在发射机和接收机没有相对运动的情况下,接收信号的多普勒频移主要受到两个方面的影响:海平面上的波浪影响和海洋内部的湍流作用。我们对这种情况进行了简化:将海面视为动态的正弦曲线,由于海面波浪的前向散射作用,声信号在传播过程中被调制,在到达接收端时发生了频率偏移,根据 Carson 定理,我们可以计算出此时接收信号的多普勒频移:

$$f_d = 2f_d \left(1 + \frac{2b \cos \theta}{c} h_b \right) \quad (2-5)$$

其中, f_b 为海面波浪运动的频率, h_b 为波浪高度的均方根值, b 为海上风速,单位为 m/s, c 为海水中的声速,一般可认为是 1500m/s, θ 为声波到达接收端的入射角。

多普勒扩展时一种由多普勒频移引起的声信号的衰落,即时间选择性衰落。假设有一单频正弦信号的频率为 f_c ,其频谱特性表现为频率 f_c 处的一条频谱轨迹,当这个单频信号被发射后,由于水深信道的多普勒效应对单频信号的影响,致使接收信号的频谱发生了展宽,即从频率为 f_c 的谱线扩展到从 $f_c - f_D$ 至 $f_c + f_D$ 的有限带宽范围内。

2.1.3 有限带宽

水声信道带宽主要由海洋中水声信号的传播损耗所决定,同时还受到水声换能器带宽限制的影响。海洋介质的弛豫吸收^[51],界面对声信号的反射、散射以及波阵面几何扩展等因素都造成了水声信号在传输过程中的衰减,严重影响了接收机的工作效率。其中海洋介质对声信号的吸收与水声信号的工作频率有关,频率

越高的信号,能量损失的就越多。其平均传播损失为:

$$TL = 20 \lg r + \alpha r \times 10^{-3} \quad (2-6)$$

上式中传播距离 r 的单位为米, α 为吸收系数 (dB/m), 与深度、温度特别是声波频率密切相关。它随着频率的增加而增加, 从而大大增加了传播损失。水下数字通信中一般的工作载波频率在 $10 \sim 1500 \text{ kHz}$ 以内, 带宽为 $200 \sim 4300 \text{ Hz}$, 而声波频率在 4000 Hz 左右为远距离传输的最佳频率。其中低于 15 kHz 为低频带, $15 \text{ kHz} \sim 150 \text{ kHz}$ 范围内为中频带, $150 \text{ kHz} \sim 1500 \text{ kHz}$ 范围内为高频带。传输距离为 $1 \sim 10 \text{ km}$ 的浅海水通信系统工作频率主要在 $10 \sim 100 \text{ kHz}$ 频率范围内; 而远距离通信中的工作频段主要在 $0 \sim 20 \text{ kHz}$ 。由此可见, 低频段的声波将在远距离水声通信中有着广泛的应用。

2.1.4 声波传播损失

由于海水中以硫酸镁和硼酸镁为主要成分的不均匀介质对信号的弛豫吸收作用、海洋环境的多变性以及声传播过程中的波阵面几何扩散, 使得声音信号在传播过程汇总损耗严重, 严重影响了接收机的接收效率。

传输损耗是指声波在海水中传播时, 由于扩散和衰减引起声波发生延迟、失真和衰落。传输损耗有两种: 衰减损失和扩散损失。

声波的衰减损失包括吸收、散射和声能漏出声道的效应。当声波频率在 1 kHz 以上时, 海水对声波的吸收是造成声波衰减的主要因素, 与声波频率的平方成正比, Thorp 等人在综合了大量的测量结果之后, 给出了海水 (含盐度 3.5%, MgSO_4 在海水溶解盐中比重为 4.7%) 对声波的吸收系数经验公式:

$$\alpha = \frac{0.11f^2}{1 + f^2} + \frac{44f^2}{4100 + f^2} \quad (2-7)$$

其中, f 为以千赫兹计的频率, α 单位: 分贝/公里。从式 (2-7) 可以看出, 海水中的声吸收系数主要取决于工作频率。从曲线变化可以看出: 随着工作频率的增加, 吸收系数是单调上升的。所以海水对声波的吸收衰落是限制水声通信工作频率的主要因素。对于中远距离来说, 一般的工作载波频率在 20 kHz 以下。

声波的扩散损失是表示当声信号从声源向外扩展时有规则减弱的几何效应。

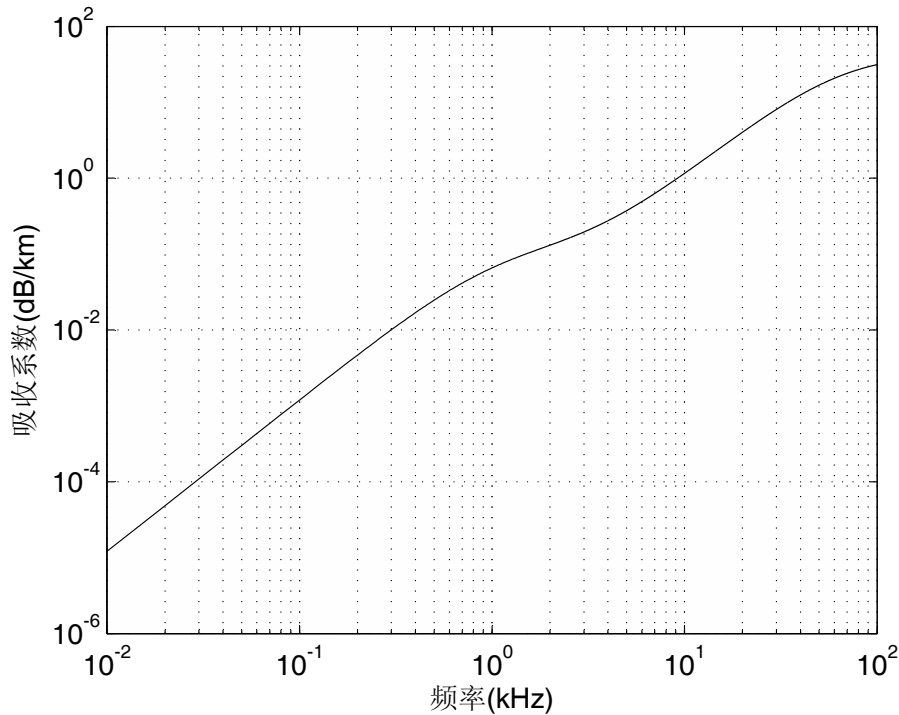


图 2-1 海水对声波的低频吸收系数

在中远距离上,随距离 R 的增加,声能量按 $\frac{1}{R^2}$ 衰减,这就是著名的反二分之三次方衰减规律。水声信道对声波的扩散损失决定了水声通信设备的最大作用距离。

2.1.5 环境噪声

在海洋中有许多噪声源,包括潮汐、湍流、海面波浪、风成噪声、生物噪声、航船以及工业噪声等,噪声的性质与噪声源有密切的关系,在不同的时间、深度和频段有不同的噪声源。在水声学中,通常用环境噪声级来描述环境噪声。

水声信道的噪声是非高斯分布的。不同的声源有着不同的带宽和噪声级,且随时间和空间变化。因此,要给出噪声的统计表达式是很困难的。实验观察可以发现,在 10Hz 以下的噪声主要来源于海洋的扰动,频率在 10Hz 到 500Hz 之间的噪声主要来源于航船和地理位置,对于较高频率噪声,即频率在 500Hz 到 50kHz 的噪声,主要来源于海面的不平整,而对于超过 50kHz 的噪声,则主要来源于海水中的分子移动。

在浅海信道,生物活动和沿岸工业也是信道噪声的来源。而且,噪声随着时

间、日期、季节、地理位置、航船密度和天气的变化将产生一个很大的变化范围。所有的这些,将使浅海信道成为一个严重的时变、空变噪声信道。

2.2 自适应均衡技术

2.2.1 自适应均衡技术综述

信道均衡是通信技术和信号处理的基本问题之一,其目的在于克服传送符号间的码间干扰,这种干扰时因为信道的非理想特性造成的,由于水声信道的特性具有时变性,因此需要自适应的调整均衡器,使得整个传输系统传输的符号间干扰被消除,这种均衡器被称为自适应均衡器。

自适应均衡过程一般包括训练和跟踪两种模式。首先,由发射机发射一系列已知的固定长度的训练序列,均衡器抽头根据训练序列作一定的调整。通常的训练序列是一串预先指定的数据或一组伪随机信号,发射机在发送训练序列后发送用户数据,经过训练的均衡器在判决引导模式完成对抽头系数的调整,对信道做出跟踪补偿。

训练序列的设计必须能够在最恶劣的信道条件下,当训练序列结束以后,能使得均衡器系数接近最优。这样,当接收用户数据时,自适应均衡器就能够跟踪信道的变化。为了保证始终有效的 ISI,需要周期的重复不断的训练均衡器。因为在数字通信中用户数据是被分为若干段放在相应地时间段中传送的,所以均衡器被大量用于数字通信中。

第一个自适应滤波器(或者自学习滤波器)通常归功于 Lucky,他曾在 1966 年为了补偿数据传输系统产生的畸变而设计了一种迫零均衡器。然而,早在 1960 年,Jakowatz 等人在自适应波形识别研究中已经报告了类似的工作。有关自适应滤波器方面的理论研究由 Giaser 和英国的 Gabor 等人在 1961 年报道,他们将它用于模拟磁带传输机构中以调整非线性“学习滤波器”的权系数。

自适应滤波器的这些早期成果许多是在各个研究机构独立的取得的。除了上面提到的,另外一些有价值的早期工作是由德国的卡尔斯卢埃技术学院和斯坦福大学完成的,他们从 1959 年起就开始研究自适应模型识别系统。1964 年,这两个单位之间的合作交流使得双方都有机会评论对方的技术,取长补短,这样就促

进了用途最广泛的调整处理器权系数的算法和研究。其相关的研究同时也在莫斯科的自动化和遥控机械研究所内展开。在 60 年代中, Rudian 发表了有关自适应滤波器进展现状的书面综述及其在自动均衡应用方面的早期参考资料。更晚些时候, Weinstein 在电话回波抵消, Qureshi 在自适应均衡方面发表了简要评述。

有许多方法可以用来调整滤波器的权系数值, 以获得最优解, 下面就对自适应均衡算法作简单介绍。

2.2.2 自适应均衡算法

自适应均衡器要采用自适应信号处理算法来调整可调参数滤波器的系数。广泛采用的算法主要有两大类^[52], 即最小均方 (LMS) 类算法和递归最小二乘 (RLS) 类算法。LMS 算法以其计算复杂度低, 结构简单而得到广泛应用。但它的收敛速度慢, 针对这个问题又研究出许多自适应 LMS 类算法, 如归一化 LMS 算法、变换域 LMS 算法、快随自优化 LMS 算法等。RLS 类算法有运算复杂度、计算量大的缺点, 但其收敛性能较好。尽管自适应算法类型很多, 但最终都源于 LMS 和 RLS 这两类算法。

LMS 算法

最小均方算法是基于最小均方误差准则的自适应算法。其基本思想是通过调整滤波器系数使均方误差最小。它由最陡下降算法推导出来。最陡下降算法是基于统计的观点 (集平均), 通过递推法得到最优权值, 但是它的主要限制是需要准确测得每次迭代的梯度矢量的方法^[52], 故只在统计平均的意义下才与最陡下降梯度下降法等效, 其解与后者相比也呈现不同程度的波动。尽管如此, LMS 算法仍以其简洁的原理和算法受到重视, 在自适应领域中占有非常重要的地位。

设输入信号为 $\mathbf{Z}_N(n) = [z(n), z(n-1), \dots, z(n-N+1)]^T$, 延迟抽头 (权系数) 为 $\mathbf{W}_N(n) = [w_0(n), w_1(n), \dots, w_{N-1}(n)]^T$, 信号的真实值为 $x(n)$, 输出 $\hat{x}(n)$ 是输入的加权值, 则:

$$\hat{x}(n) = \mathbf{W}_N^H(n) \mathbf{Z}_N(n) \quad (2-8)$$

$$e(n) = x(n) - \hat{x}(n) \quad (2-9)$$

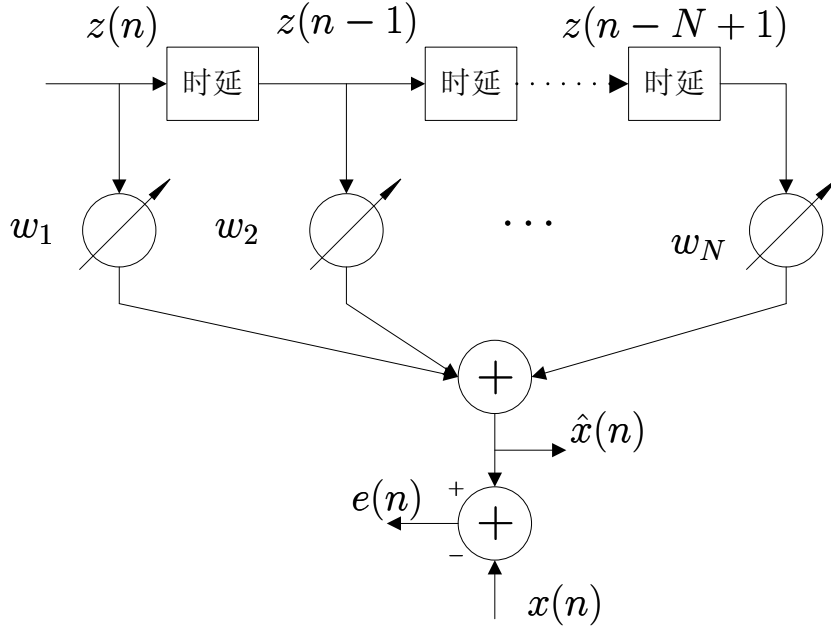


图 2-2 基本 LMS 算法框图

$$J(n) = E(|e(n)|^2) \quad (2-10)$$

$$\mathbf{W}_N(n+1) = \mathbf{W}_N(n) - \mu \nabla(J(n)) \quad (2-11)$$

其中,上标 H 表示共轭转置。

由式 (2-11) 可见,要得到 LMS 算法的更新公式,须确定 $\nabla(J(n))$ 及合适选择步长因子 μ 。

我们可以按照下面的方法得到 $\nabla(J(n))$ 的估计值:

$$\hat{\nabla}(J(n)) = \nabla(|e(k)|^2) = -2e^*(n)\mathbf{Z}_N(n) \quad (2-12)$$

可以证明梯度的估计值是无偏估计,即:

$$E\{\hat{\nabla}(J(n))\} = \nabla(J(n)) \quad (2-13)$$

将式 (2-12) 代入式 (2-11) 得:

$$\mathbf{W}_N(n+1) = \mathbf{W}_N(n) - \mu \mathbf{Z}_N(n)e^*(n) \quad (2-14)$$

式 (2-8)、(2-9) 和 (2-14) 共同构成了 LMS 算法。对于 μ 的选择,须满足:

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (2-15)$$

λ_{\max} 是输入信号自相关矩阵 R_x 的最大特征值。步长因子 μ 对算法收敛过程有很大的影响,过大或过小都是不合适的。

当信道特征稳定时,存在一个最优的 μ 值使 $\mathbf{W}_N(n)$ 经过有限次迭代后达到最优。但是,对于起伏较大的信道,固定 μ 值的 LMS 算法就不能取得好的结果。若 μ 值能够随信道特性自适应变化,便可改善 LMS 算法的信道跟踪性能,于是便有了快速自优化 LMS(FOLMS) 算法。

快速自优化 LMS 算法 (FOLMS)

LMS 算法仅有一个控制因子 μ ,LMS 算法的特性主要取决于 μ 值的选取,为了保证算法收敛,通常需要把 μ 取得较小,但是有时要求快速收敛,又需要将 μ 值取的尽量大一点。因此实际应用汇总 μ 的选取往往很困难,其取值直接影响了算法的收敛及跟踪速度和稳定性。

考虑到在每步更新过程中,肯定存在一个最优的 μ ,使权系数 $\mathbf{W}_N(n)$ 在更新时能达到瞬时最优,如果 μ 已经固定,LMS 算法就不可能有非常好的效果。此时应当考虑使 μ 能够随系统的变化而自适应地变化的方法,快速自优化 LMS 算法 (Fast Self-Optimized LMS),简称 FOLMS 算法,就是一种快速的自优化 LMS 算法^[53--55]。

正如前面所说,对于 LMS 算法来说,算法每次更新自适应滤波器权系数 $\mathbf{W}_N(n)$ 时一定存在一个最优的 μ ,使得 MSE 最小。因此,推导 FOLMS 算法的基本思想是通过 MSE 对 μ 求导数,写出 μ 的递推更新公式,使之随着 MSE 的变化自适应调节,逐步地最小化均方误差。这样就可以肯定取得比 LMS 算法更好的跟踪时变的能力。

由前面可知 MSE 的代价函数为:

$$J(n) = E(|e(n)|^2) \quad (2-16)$$

式中:

$$e(n) = x(n) - \mathbf{W}_N(n)^H \mathbf{Z}(n) \quad (2-17)$$

将式 (2-16) 对 μ 求导,得到 $J(n)$ 对 μ 的梯度 $\nabla_{\mu}(n)$:

$$\nabla_{\mu}(n) = \frac{\partial J(n)}{\partial \mu} = E \left(\frac{\partial e(n)}{\partial \mu} e^*(n) + \frac{\partial e^*(n)}{\partial \mu} e(n) \right) \quad (2-18)$$

由式 (2-17) 可得:

$$\frac{\partial e(n)}{\partial \mu} = -\Psi^H \mathbf{Z}(n) \quad (2-19)$$

上式中, $\Psi(n)$ 表示权向量 $\mathbf{W}_N(n)$ 对 μ 的梯度:

$$\Psi(n) = \frac{\partial \mathbf{W}(n)}{\partial \mu} \quad (2-20)$$

因此式 (2-18) 可以改写为:

$$\begin{aligned} \nabla_{\mu}(n) &= -E(\Psi^H(n) \mathbf{Z}(n) e^*(n) + \mathbf{Z}^H(n) \Psi(n) e(n)) \\ &= 2\text{Re}[\Psi^H(n) \mathbf{Z}(n) e^*(n)] \end{aligned} \quad (2-21)$$

其中的 $\text{Re}[\cdot]$ 表示对复数取实部。

借助于 LMS 算法中用瞬时梯度代替平均梯度的思想,我们也可以得出 μ 值的更新递推算法:

$$\mu(n+1) = \mu(n) - \alpha \nabla_{\mu}(n) \quad (2-22)$$

其中的 α 是一个很小的常数。

整个 FOLMS 算法可以归纳如下:

$$\begin{aligned} \hat{x}(n) &= \mathbf{W}_N(n)^H \mathbf{Z}(n) \\ e(n) &= x(n) - \hat{x}(n) \\ \mathbf{W}_N(n+1) &= \mathbf{W}_N(n) + \mu(n) \mathbf{Z}(n) e^*(n) \\ \mu(n+1) &= [\mu(n) - \alpha \text{Re}[\Psi^H(n) \mathbf{Z}(n) e^*(n)]]_{\mu_{\min}}^{\mu_{\max}} \\ \Psi(n+1) &= [\mathbf{I} - \mu(n) \mathbf{Z}(n) \mathbf{Z}^H(n)] \Psi(n) + \mathbf{Z}(n) e^*(n) \end{aligned} \quad (2-23)$$

从上式可以看出, FOLMS 算法和 LMS 算法类似,只是多了更新 μ 这个环节,该算法在每次更新时大约需要 $4N$ 次的乘法运算,其中 N 为滤波器的阶数。

需要特别指出的是,为了保证算法的稳定性,在实际应用中,更新 μ 的步骤中需要对其设定门限值 μ_{\max} 和 μ_{\min} 。 μ_{\min} 取一接近于零的正值即可,而 μ_{\max} 的选取对算法的性能至关重要,需要仔细选取^[56]。

RLS 算法

LMS 算法尽管简单易懂,但由于其中只有一个可调量 μ ,性能在一定程度上受到限制,收敛速度慢,收敛所需码元数较多,用以各时刻的抽头参量(即权值)等

作该时刻数据块估计时平方误差为最小的准则,而未用现时刻的抽头参量等来对以往各时刻的数据块作重新估计后的累计平均误差为最小的准则,故对非平稳信号的适应性差,只有在稳态时才能达到最优解。为了克服上述缺点,我们采用新的准则:在每一时刻对所有已输入信号二重估的平方误差和最小的准则,即最小二乘准则。这一准则在现有的约束条件下,利用了最多可利用的信息,是更为有效的,对信号的非平稳性适应能力也较 LMS 准则好。下面,我们还是先简单地推导其算法,推导过程中与前面意义相同或相似的量不再重复加以说明。

递归最小二乘 (Recursive Least Square, RLS) 法所遵循的准则是确定 \mathbf{W}_N , 使 $e(i|n) = x(i) - \mathbf{Z}'_N(i)\mathbf{W}_N(n)$ 的加权平方和为最小。

$$\varepsilon(n) = \sum_{i=1}^N \lambda^{n-i} e^2(i|n) \quad (2-24)$$

其中, λ 被称为遗忘因子, 略小于 1, 通常的取值在 0.95~0.9995 之间, λ^{n-i} 因子的物理含义有: 在该准则内所用到的各输入信号中添加指数权, 即对靠近当前时刻的数据加以较大的权来考虑, 而时间靠前的数据, 其权按指数规律逐渐减少, 这样可使算法更能反映当前情况, 从而加强对非平稳信号的适应性。

满足 $\frac{\partial}{\partial \mathbf{W}_N(n)=0}$ 的 $\mathbf{W}_N(n)$ 值为

$$\mathbf{W}_N(n) = \mathbf{R}_{NN}^{-1}(n) \mathbf{p}_N(n) \quad (2-25)$$

式中,

$$\mathbf{R}_{NN}^{-1}(n) = \sum_{i=1}^N \lambda^{n-i} \mathbf{Z}_N(i) \mathbf{Z}_N'(i) \quad (2-26)$$

$$\mathbf{p}_N(n) = \sum_{i=1}^N \lambda^{n-i} x(i) \mathbf{Z}_N(i) \quad (2-27)$$

我们最终要导出的是一个由 $n-1$ 时刻的各量及现时刻 n 的输入数据 $z(n)$ 和 $x(n)$ 表示的便于求解的迭代式, 由于在推导过程中要用到矩阵论中的定理, 可以参考 [52], 在此为节省篇幅, 以突出重点, 只给出推导结果。

$$\mathbf{W}_N(n+1) = \mathbf{W}_N(n) + \mathbf{g}_N(n)e(n) \quad (2-28)$$

式中, $\mathbf{g}_N(n)$ 为增益向量,

$$\mathbf{g}_N(n) = \frac{\mathbf{C}_{NN}(n-1)\mathbf{z}_N(n)}{\lambda + \mu(n)} \quad (2-29)$$

$$\mathbf{C}_{NN}(n) = \mathbf{R}_{NN}^{-1}(n) = \frac{1}{\lambda} [\mathbf{C}_{NN}(n-1) - \mathbf{g}_N(n)\mathbf{Z}'(n)\mathbf{C}_{NN}(n-1)] \quad (2-30)$$

$$\mu(n) = \mathbf{Z}'(n)\mathbf{C}_{NN}(n-1)\mathbf{z}_N(n) \quad (2-31)$$

我们注意到, RLS 算法中 $\mathbf{g}_N(n)$ 的因子 $\frac{\mathbf{C}_{NN}(n-1)}{\lambda + \mu(n)}$ 与 LMS 算法中的 μ 作用相似, 但 μ 是一个标量, 而该因子则是随时刻 n 而调整的矩阵, 这说明在不同时刻, $\mathbf{W}_N(n)$ 的各元素均随更新的输入数据以不同的步长做调整, 而不是统一地以同一个因子来调整, 表征了调整的精细性及新信息数据利用的充分性。另外, 该算法的准则综合考虑了该时刻以前的全部信息, 使得在收敛过程的每一点得到的都是最优解。但从 RLS 算法的迭代式可以看出, 其码元间的计算量 $\propto N^2$, 而 LMS 算法 $\propto N$, 可见, RLS 法是以增加计算量为代价换取好的收敛性能的。

式 (2-30) 给出的 $\mathbf{C}_{NN}(n)$ 的递归更新方程数值稳定性不好, 因此人们研究出数值稳定性好的平方根 RLS 算法, 它是基于 $\mathbf{C}_{NN}(n)$ 的平方根因式分解, 即 $\mathbf{C}_{NN}(n) = \mathbf{U}(n)\mathbf{D}(n)\mathbf{U}'(n)$, 其中 $\mathbf{U}(n)$ 是上三角矩阵, $\mathbf{D}(n)$ 是对角矩阵, 这些算法直接更新 $\mathbf{U}(n)$ 而不计算 $\mathbf{C}_{NN}(n)$, 但是它的复杂度仍然正比与 N^2 。

RLS 算法也有快速算法, 1991 年 T. Kailath, D. Slock^[57--61] 等人提出了 RLS 快速算法, 即快速横向滤波器 (SFTF) 算法。SFTF 的运算量为 $8N$, 与 RLS 算法相比有大幅的减少。

分集合并自最优自适应多通道判决反馈均衡算法

消除由于水声信道多径产生的码间干扰 (ISI), 是建立可靠、高速水声通信系统面临的一个主要问题。目前, 解决该问题的主要手段是结合空间分集技术的自适应判决反馈均衡器 (DFE), 即多通道判决反馈均衡器 (MC-DFE)^[56, 62--70]。自适应判决反馈均衡器按照最小化均方误差 (MMSE) 准则自适应调节前向和反馈滤波器的权系数, 消除码间干扰。因此, 随着信道多径的扩展, 均衡器的权系数的个数也要相应增加, 从而更好地消除码间干扰, 保证数据的可靠传输, 但这同时增加了运算复杂性。此外, 多通道处理的运算复杂性明显高于单通道处理。

文献[71,72]提出了一种基于 RLS 算法的自适应分集合并器(RLSDC),优化多通道判决反馈均衡器的性能。由于 RLS 算法使用输入信号的自相关矩阵的逆(该逆矩阵在每次迭代的时候自适应更新)对输入信号进行解相关运算,因此运算也很复杂。此外,接收结构中的载波相位估计器,文献[71]使用的是锁相环(PLL),该方法中的相位跟踪因子为两个固定常数,不能自适应更新。

为了进一步优化算法,提高系统性能,文献[73]在[74]的基础上,对接收算法进行了优化,提出了一种运算量更小,但性能更优的接收算法。和文献[71]算法相比,本文的均衡算法有以下三个特点:

1. 采用快速自优化 LMS 分集合并(FOLMSDC)算法对合并器系数进行更新。LMS 算法不计算相关函数,也不用矩阵求逆,具有较低的运算复杂性;
2. FOLMSDC 算法采用变步长因子算法,该算法一方面可以改善传统 LMS 算法的收敛速度,另一方面具有更好的信道跟踪性能;
3. 载波相位估计采用快速自优化 LMS 相位补偿(FOLMSPC)算法。该算法中的相位跟踪因子可以按照最小化均方误差(MMSE)准则自适应地更新,从而更好地校正相位失调,性能明显优于锁相环(PLL)技术。

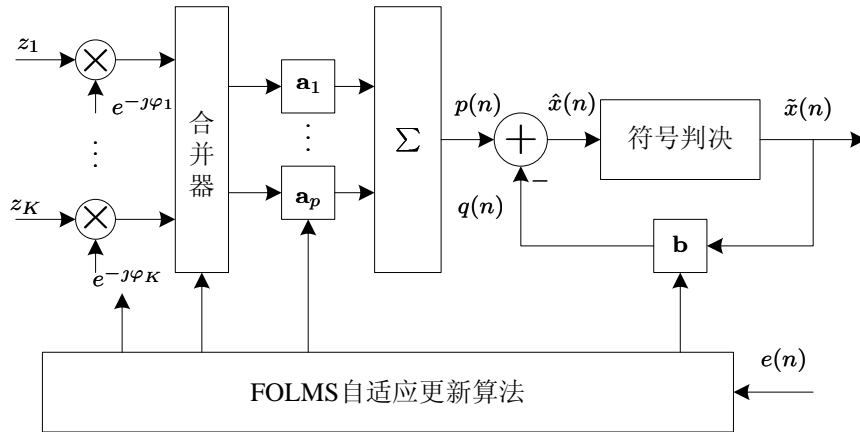


图 2-3 分集合并自适应最佳自适应多通道判决反馈均衡器框图

文献[73]中详细推导了该算法的各个步骤,这里不再赘述,仅给出该算法与[71]中算法的两个比较:

- 在算法性能方面,无论是低信噪比还是高信噪比,本文提出的均衡算法(FOLMSDC+FOLMSPC+FOLMS)的性能都要优于文献[71]的算法(RLSDC+SFTF+PLL)。本文提出的优化分集合并器能够对输入的 K 路信号优化加权,变成最优的 P 路信号,一方面减少了处理复杂性,另一方面优化了系统的接收性能。与此同时,FOLMSDC 算法一方面改善了传统 LMS 算法的收敛性能,另一方面它的信道跟踪性能优于 LMS 算法和 RLS 算法。FOLMSPC 算法的相位跟踪因子可以自适应的更新,其相位校正能力比常数因子的 PLL 算法有明显提高。
- 在运算量方面,文献[73]中提出的均衡算法(FOLMSDC+FOLMSPC+FOLMS)的运算量小于文献[71]的算法(RLSDC+SFTF+PLL)。未使用合并器算法的 K 通道判决反馈均衡器在单个符号均衡时需要更新 $K \times N$ 个权系数,而使用合并器算法 $K \sim P$ 通道判决反馈均衡器需要更新的权系数个数为 $K \times P + P \times N = (K + P) \times N$ 。此外,RLS 的运算量为 $2N^2 + 6N$,其快速算法(SFTF)的运算量也要 $8N$,而 LMS 的运算量为 $2N$,FOLMSDC 算法的运算量也仅为 $4N$ 。一般来讲, N 的取值都在十阶以上。在 $K = 8, P = 4, N = 16$ 的情况下,FOLMSDC+FOLMSPC+FOLMS 算法的运算量最小,FOMSPC+FOLMS 算法的运算量次之,RLSDC+SFTF+PLL 算法的运算量最大。由此可见,文献[73]中的均衡算法的运算量与不使用 FOLMSDC 算法的均衡算法(FOLMSPC+FOLMS)和文献[71](RLSDC+SFTF+PLL)的算法相比,运算复杂度是最小的。

2.3 Turbo 均衡

传统的均衡方式有线性均衡、判决反馈均衡、最大似然序列估计等。这些均衡方式在一定程度上克服了符号间干扰带来的影响,但是这些均衡方式中,均衡与解码时独立进行的,解码器对接收来的均衡器的信息进行解码。由于这种结构本身的特性,使得它对均衡器判决后的突发错误无法很好地纠正,也无法利用译码可靠信息,因而均衡的效果不太理想。而 Turbo 均衡利用了 Turbo 码译码算法中提出的迭代思想,将均衡和译码很好地结合起来,在均衡器和译码器之间反复迭

代可靠性信息,提高了均衡译码的整体性能,是一种联合均衡和译码技术。把这种联合均衡和译码技术引入到实际通信系统中可以提高系统传输的可靠性。

2.3.1 Turbo 均衡原理

二进制数据 a_n 经过卷积编码器并经过符号映射,则 $c_n = [c_{n,1}, c_{n,2}, \dots, c_{n,Q}]$, $c_{n,j} \in \{-1, +1\}$, $n = 1, 2, \dots, K_c$, 经过交织结合训练序列 t_n 之后, $x_n = \Pi(c_n)$, 其中 $\Pi(\cdot)$ 表示交织, $\Pi^{-1}(\cdot)$ 表示解交织。

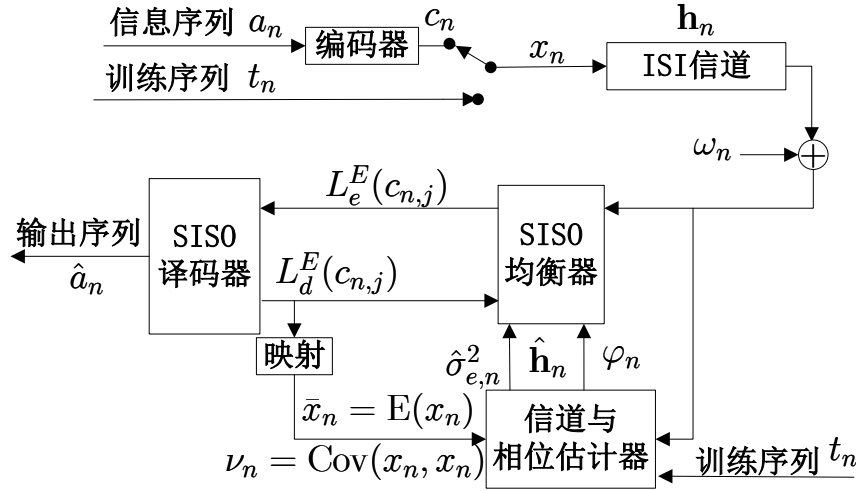


图 2-4 Turbo 均衡水声相干通信系统传输模型

设信道噪声是零均值的加性高斯白噪声 (AWGN), 则噪声抽样 ω_n 是满足独立同分布 (i.i.d):

$$f_{\omega}(x) = N(0, \sigma_{\omega}^2) \quad (2-32)$$

其中 $N(\mu, \sigma^2) \triangleq \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$, μ 表示随机变量的均值, σ^2 表示方差。

设接收到的符号序列为 $\mathbf{z} = [z_1, z_2, \dots, z_{K_c}]^T$, 则

$$z_n = \sum_{k=-M_1}^{M_2} h_k x_{n-k} + \omega_n \quad (2-33)$$

对于 SISO 均衡器计算的是:

$$L_e^{\text{Turbo}}(c_{n,j}) = \ln \frac{p(c_{n,j} = +1|\mathbf{z})}{p(c_{n,j} = -1|\mathbf{z})} - \ln \frac{p(c_{n,j} = +1)}{p(c_{n,j} = -1)} = L^{\text{Turbo}}(c_{n,j}) - L_a^{\text{Turbo}}(c_{n,j}) \quad (2-34)$$

在均衡器输出的 $L_e^{\mathrm{mathrmE}}(c_{n,j})$ 是外部信息,是由接收序列 \mathbf{z} 与其它时刻的先验信息获取的,并不受现时刻先验信息 $L_a^{\mathrm{mathrmE}}(c_{n,j})$ (有 SISO 解码器输出)的影响,因而经解交织后可用作解码器的先验信息。

对于 MAP 译码器,在给定的输入序列为 $\mathbf{r} = L_1(c_{n,j})$ 时,计算的是

$$\begin{aligned} L^{\mathrm{mathrmD}}(c_n) &= \ln \frac{p(c_{n,j} = +1|\mathbf{r})}{p(c_{n,j} = -1|\mathbf{r})} \\ &= \ln \frac{p(\mathbf{r}|c_{n,j} = +1)}{p(\mathbf{r}|c_{n,j} = -1)} + \ln \frac{p(c_{n,j} = +1)}{p(c_{n,j} = -1)} \\ &= L_e^{\mathrm{mathrmD}}(c_{n,j}) + L_a^{\mathrm{mathrmD}}(c_{n,j}) \end{aligned} \quad (2-35)$$

其中 $L_a^{\mathrm{mathrmD}}(c_{n,j})$ 为先验信息, $L_e^{\mathrm{mathrmD}}(c_{n,j})$ 为外部信息,解码器的外部信息 $L_e^{\mathrm{mathrmD}}(c_{n,j})$ 经过交织后可以作为先验信息反馈给 SISO 均衡器, SISO 均衡器利用此信息再次运算,输出比前一次更准确的似然比,经过几次迭代运算,输出的误码率大为减少。

2.3.2 Turbo 均衡算法

现在较为常用的 Turbo 均衡算法由 MAP 均衡算法、基于 MMSE 准则的判决反馈均衡算法 (MMSE-DFE) 以及基于 MMSE 准则的线性均衡算法 (MMSE-LE)。MAP 均衡算法的性能最好,因为它是基于使码元误码率最小的算法,但是复杂度随着信道长度 M 呈指数增长,所以算法的复杂度极高,不利于实现。然后就研究了基于 MMSE 准则的均衡算法来降低复杂度。下面从两个方面对这三种算法进行比较。

算法复杂度

以下给出了不同均衡算法的复杂度比较,其中, N 为均衡滤波器的长度, M 为信道响应的长度, 2^m 为发送信号星座图中字符集大小。

表 2-1 算法复杂度比较

算法	MAP	MMSE-LE	MMSE-DFE
复杂度	$O(2^{mM})$	$O(M^2 + N^2)$	$O(M^2 + N^2)$

从表2-1可以看出,MAP 算法的复杂度最高,呈指数型增长,因此在实际应用

中很难实现。而 MMSE-DFE 算法和 MMSE-LE 算法的复杂度一样,因此需要根据其均衡性能来决定哪种算法更优。

算法的性能

选择 Proakis'B 信道作为仿真信道,其信道冲击响应为:

$$h_B(n) = 0.407\delta(n+1) + 0.815\delta(n) + 0.407\delta(n-1) \quad (2-36)$$

信道编码采用码率 $R = 1/2$ 的递归系统卷积码,生成多项式为 $[7, 5]$,交织器采用 $S = 16$ 的随机交织器,译码器均采用 MAP 译码,帧长为 640,迭代次数为 6 次。基于 MMSE-LE 和 MMSE-DFE 的滤波器参数为: $N_1 = 10, N_2 = 10$ 和 $N = 21, N_b = 2$ 。信噪比 SNR 的定义为:

$$SNR = 10 \log(E_z/N_0 R) = 10 \log(1/2\sigma_\omega^2 R)dB \quad (2-37)$$

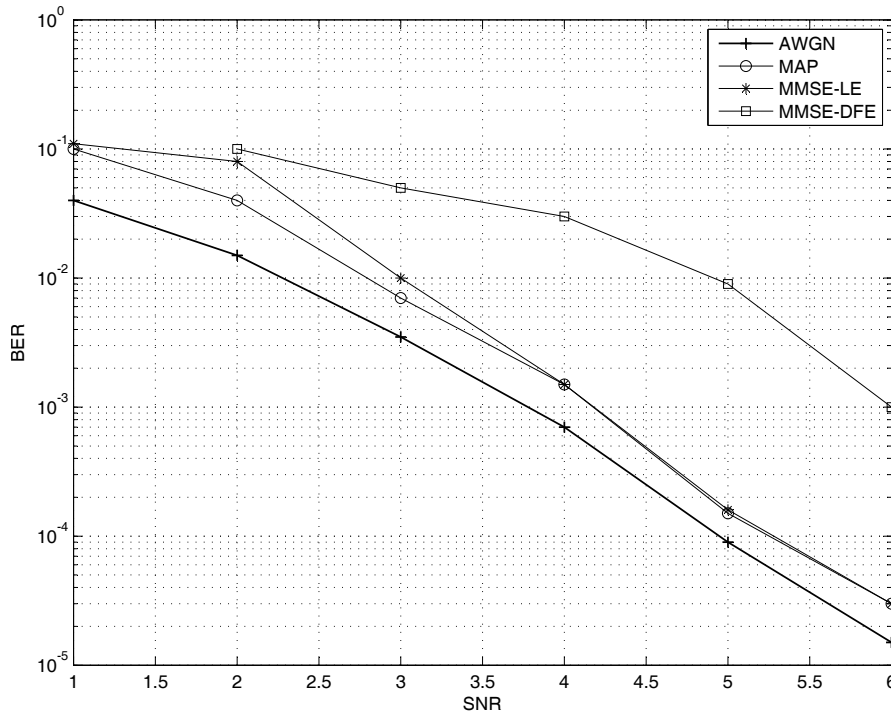


图 2-5 不同 Turbo 均衡算法的性能比较

图2-5是几种算法的性能比较,因为 MAP 均衡性能采用的是最优准则,在高信噪比下,经过迭代可以达到 AWGN 下的性能,所以性能最好,但同时复杂度也最

高,不适合在实际系统中应用。MMSE-LE 均衡器性能仅次于 MAP 均衡器,尽管传统的 DFE 比 LE 好,但是 Turbo 均衡应用中,基于先验信息的 DFE 均衡器的性能比线性 MMSE 均衡器差,而运算复杂度方面,二者是一样的。因此,在水声通信系统中,本文采用基于先验信息 MMSE 准则的线性 Turbo 均衡算法,并在下一章详细介绍。

2.4 本章小节

本章对海洋声学信道传输特性及其对水声通信的影响、水声信道的衰落特性及其水声多径信道模型进行了介绍和简单的分析,为水声通信系统的设计、技术参数的选取提供了重要依据。本章介绍了 LMS 算法及其快速自优化算法 (FOLMS) 和 RLS 算法及其快速算法 (SFTF),并给出了这两类算法的优缺点。在此基础上文献[73]提出了一种适用于水声相干通信系统的“分集合并自最佳自适应多通道判决反馈均衡算法”,即 FOLMSDC+FOLMSPC+FOLMS 算法,最后并与[71]作比较,在性能和运算量上都有很大的优势。

另外介绍了与传统均衡算法结构不同的 Turbo 均衡技术,并对其中的几种均衡算法的运算复杂度以及性能加以比较,得出适用于水声通信系统的基于先验信息 MMSE 准则的线性 Turbo 均衡算法。

第三章 基于先验信息 MMSE 准则的线性 Turbo 均衡算法

3.1 MMSE 均衡算法

如图3-1为水声相干通信系统的传输模型,包括发射部分,传输部分以及接收部分。在接收部分中又包含了均衡器、译码器以及信道估计器。本章的重点在于均衡算法,因此主要关注 SISO 均衡器。

现在把图2-4中的接收部分单独拿出来,并采用基于先验信息 MMSE 准则的线性 Turbo 均衡器 (MMSE-LE) 代替图2-4中的 SISO 均衡器。如图3-1

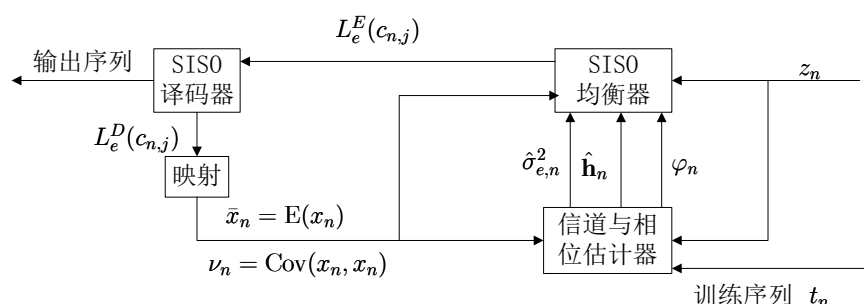


图 3-1 基于 MMSE-LE 的接收端

从图3-1中可以看出,接收符号 z_n 以及 SISO 译码器的软输出作为 MMSE-LE 的输入,并给 SISO 译码器提供软信息,从而构成一个回路,通过不断的迭代来提高均衡和译码的性能。为了专注于均衡算法部分,将图3-1中与均衡无关的部分暂且去除,从而得到图3-2

从图3-2中可以看出,MMSE-LE 包括两个部分,

1. MMSE 估计器
2. 映射

下面就从这两个方面对 MMSE-LE 算法进行详细介绍。

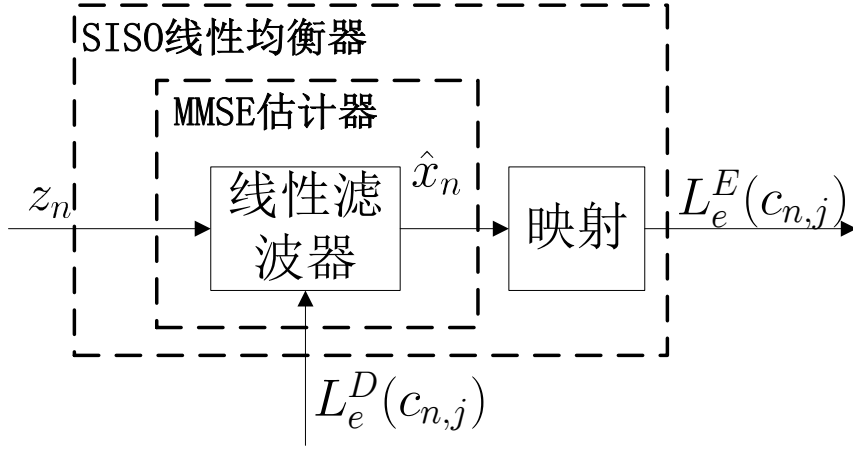


图 3-2 MMSE-LE 内部结构框图

3.1.1 MMSE 估计器

利用长度为 $N = N_1 + N_2 + 1$ 的接收符号序列 $\mathbf{z}_n = [z_{n-N_2}, z_{n-N_2+1}, \dots, z_{n+N_1}]$ 来计算发送符号 x_n 的线性估计值 \hat{x}_n :

$$\hat{x}_n = \mathbf{a}_n^H \mathbf{z}_n + b_n \quad (3-1)$$

其中 $\mathbf{a}_n \triangleq [a_{n,N_2}^*, a_{n,N_2-1}^*, \dots, a_{n,-N_1}^*]^T$, $a_{n,k} \in \mathbb{C}$ 为线性滤波器的系数, $b_n \in \mathbb{C}$ 是在给定先验信息的情况下, 随机变量 x_n 的非零均衡的时变偏移补偿。 N_1 和 N_2 为线性滤波器的非因果和因果长度。 N 为滤波器的总长度。

当允许 \mathbf{a}_n 和 b_n 都随 n 变化时, 通过最小均方误差准则 (MMSE), 选择

$$\begin{aligned} \mathbf{a}_n &= \text{Cov}(\mathbf{z}_n, \mathbf{z}_n)^{-1} \text{Cov}(\mathbf{z}_n, x_n) \\ b_n &= E(x_n) - \mathbf{a}_n^H E(\mathbf{z}_n) \end{aligned} \quad (3-2)$$

来最小化 $E(|x_n - \hat{x}_n|^2)$, 从而得出:

$$\hat{x}_n = E(x_n) + \text{Cov}(x_n, \mathbf{z}_n) \text{Cov}(\mathbf{z}_n, \mathbf{z}_n)^{-1} (\mathbf{z}_n - E(\mathbf{z}_n)) \quad (3-3)$$

这种算法称作为最小均方误差算法 (MMSE)。

上式中, \mathbf{z}_n 为接收符号序列:

$$\mathbf{z}_n = \mathbf{H}_n \mathbf{x}_n + \boldsymbol{\omega}_n \quad (3-4)$$

其中, $\mathbf{x}_n \triangleq [x_{n-N_2-M+1}, x_{n-N_2-M+2}, \dots, x_{n+N_1}]^T$ 为发送符号序列, H 为 $N \times (N + M - 1)$ 的信道矩阵, M 为信道冲击响应长度:

$$\mathbf{H}_n \triangleq \begin{bmatrix} h_{M-1} & h_{M-2} & \cdots & h_0 & 0 & \cdots & 0 \\ & h_{M-1} & h_{M-2} & \cdots & h_0 & 0 & \cdots & 0 \\ & & & \ddots & & & & \\ 0 & & \cdots & 0 & h_{M-1} & h_{M-2} & \cdots & h_0 \end{bmatrix} \quad (3-5)$$

ω_n 为均值为 0, 方差为 σ_n^2 的高斯白噪声。

假设噪声 ω_n 是独立同分布的 (i.i.d), 可以得到:

$$\begin{aligned} E(\mathbf{z}_n) &= \mathbf{H}_n E(\mathbf{x}_n) \\ \text{Cov}(x_n, \mathbf{z}_n) &= \text{Cov}(x_n, x_n) [\mathbf{0}_{1 \times (N_2+M-1)} \quad 1 \quad \mathbf{0}_{1 \times N_1}] \mathbf{H}_n^H \\ \text{Cov}(\mathbf{z}_n, \mathbf{z}_n) &= \sigma_{\omega,n}^2 \mathbf{I}_N + \mathbf{H}_n \text{Cov}(\mathbf{x}_n, \mathbf{x}_n) \mathbf{H}_n^H \end{aligned} \quad (3-6)$$

由于比特 $c_{n,j}$ 是独立同分布的, 进而 x_n 也是独立的, 从而可以得出, 对于 $n, m, n \neq m$ 的情况下, $\text{Cov}(x_n, x_m) = 0$ 。所以协方差矩阵 $\text{Cov}(\mathbf{x}_n, \mathbf{x}_n)$ 只有在主对角线上式非零值。

如图3-1, 为了计算法 \mathbf{a}_n, b_n , 利用译码器反馈软信息 $L_e^D(c_{n,j})$ 可得随机变量 x_n 的先验信息和方差为:

$$\begin{aligned} \bar{x}_n &\triangleq E(x_n) = \sum_{\alpha_i \in \mathcal{S}} \alpha_i \cdot P(x_n = \alpha_i) \\ v_n &\triangleq \text{Cov}(x_n, x_n) = \left(\sum_{\alpha_i \in \mathcal{S}} |\alpha_i|^2 \cdot P(x_n = \alpha_i) \right) - |\bar{x}_n|^2 \end{aligned} \quad (3-7)$$

式3-5中概率 $P(c_{n,j})$ 是对数似然比 $L_e^D(c_{n,j})$ 的函数,

$$\begin{aligned} P(x_n = \alpha_i) &= \prod_{j=1}^Q P(c_{n,j} = s_{i,j}) \\ &= \prod_{j=1}^Q 1/2 \cdot (1 + \tilde{s}_{i,j} \cdot \tanh(L_e^D(c_{n,j}/2))) \end{aligned} \quad (3-8)$$

其中

$$\tilde{s}_{i,j} \triangleq \begin{cases} +1 & s_{i,j} = 0 \\ -1 & s_{i,j} = 1 \end{cases} \quad (3-9)$$

表 3-1 QPSK 映射方式

i	1	2	3	4
$s_{i,1} s_{i,2}$	00	10	01	11
α_i	$(+1 + \imath)/\sqrt{2}$	$(-1 + \imath)/\sqrt{2}$	$(+1 - \imath)/\sqrt{2}$	$(-1 - \imath)/\sqrt{2}$

表 3-2 从 $L_e^D(c_{n,i})$ 求解 \bar{x}_n 和 v_n

QPSK:
$\bar{x}_n = 1/\sqrt{2} \cdot (\tanh(L_e^D(c_{n,1})/2) + \tanh(L_e^D(c_{n,2})/2)\imath)$
$v_n = 1 - \bar{x}_n ^2$

如果采用表3-1的映射方式,那么, \bar{x}_n 和 v_n 的求解可以简化为表3-2

利用 \bar{x}_n 和 v_n 给出以下定义:

$$\begin{aligned}
\bar{z}_n &\triangleq E(z_n) = \sum_{k=0}^{M-1} h_k \bar{x}_{n-k} \\
\bar{\mathbf{z}}_n &\triangleq E(\mathbf{z}_n) = [\bar{z}_{n-N_2}, \bar{z}_{n-N_2+1}, \dots, \bar{z}_{n+N_1}]^T \\
\mathbf{V}_n &\triangleq \text{Cov}(x_n, x_n) = \text{diag}(v_{n-M-N_2+1}, \dots, v_{n+N_1}) \\
\mathbf{s}_n &\triangleq \mathbf{H}_n [\mathbf{0}_{1 \times (N_2+M-1)} \quad 1 \quad \mathbf{0}_{1 \times N_1}]^T \\
\Sigma_n &\triangleq \text{Cov}(\mathbf{z}_n, \mathbf{z}_n) = (\sigma_\omega^2 \mathbf{I}_N + \mathbf{H}_n \mathbf{V}_n \mathbf{H}_n^H)
\end{aligned} \tag{3-10}$$

因此,估计值 \hat{x}_n 如下:

$$\hat{x}_n = \bar{x}_n + \mathbf{a}_n^H (\mathbf{z} - \bar{\mathbf{z}}_n) = \bar{x}_n + \sum_{k=-N_1}^{N_2} a_{n,k} \cdot (z_{n-k} - \bar{z}_{n-k}) \tag{3-11}$$

其中 $\mathbf{a}_n = v_n \Sigma_n^{-1} \mathbf{s}_n$, 这个方程等价于 $\mathbf{z}_n - \bar{\mathbf{z}}_n$ 经过一个线性滤波器,其滤波系数 $f_{n,k}$, $k = -N_1, 1 - N_1, \dots, N_2$, 具体如下:

$$\mathbf{f}_n = [f_{n,N_2}^*, f_{n,N_2-1}^*, \dots, f_{n,-N_1}^*]^T \triangleq \Sigma_n^{-1} \mathbf{s}_n \tag{3-12}$$

乘以 v_n 并加上 \bar{x}_n 可以得到估计符号如下:

$$\hat{x}_n = \bar{x}_n + v_n \cdot \mathbf{f}_n^H (\mathbf{z}_n - \bar{\mathbf{z}}_n) \tag{3-13}$$

然而, \hat{x}_n 通过 \bar{x}_n 和 v_n 依赖 $L_e^D(c_{n,j})$, 这违背了求解外部信息时不能依赖先验信息的原则。为了使 \hat{x}_n 独立于 $L_e^D(c_{n,j})$, 在计算 \hat{x}_n 的时候, 将 $L_e^D(c_{n,j})$, $j = 1, \dots, Q$ 都设置为 0, 从而 $\bar{x}_n = 0$ 且 $v_n = 1$ 。式 3-13 可以改写如下:

$$\begin{aligned}\mathbf{f}'_n &\triangleq \mathbf{f}_n|_{v_n=1} = (\Sigma_n + (1 - v_n)\mathbf{s}_n\mathbf{s}_n^H)^{-1}\mathbf{s}_n \\ \hat{x}_n &= 0 + 1 \cdot \mathbf{f}'_n^H(\mathbf{z}_n - \bar{\mathbf{z}}_n + (\bar{x}_n - 0)\mathbf{s}_n)\end{aligned}\quad (3-14)$$

利用逆矩阵定理, 可以得知 \mathbf{f}'_n 是 \mathbf{f}_n 的伸展版, 可以表示如下:

$$\begin{aligned}\mathbf{f}'_n &= (\Sigma_n^{-1} - \Sigma_n^{-1}\mathbf{s}_n((1 - v_n)^{-1} + \mathbf{s}_n\Sigma_n^{-1}\mathbf{s}_n)^{-1}\mathbf{s}_n^H\Sigma_n^{-1})\mathbf{s}_n \\ &= \mathbf{f}_n - \mathbf{f}_n((1 - v_n)^{-1} + \mathbf{f}_n^H\mathbf{s}_n)^{-1}\mathbf{f}_n^H\mathbf{s}_n \\ &= (1 + (1 - v_n)^{-1} + \mathbf{f}_n^H\mathbf{s}_n)^{-1}\mathbf{f}_n\end{aligned}\quad (3-15)$$

因此, 估计符号 \hat{x}_n 可以改写如下:

$$\hat{x}_n = K_n \cdot \mathbf{f}'_n^H(\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n\mathbf{s}_n) \quad (3-16)$$

其中, $K_n \triangleq (1 + (1 - v_n)\mathbf{f}_n^H\mathbf{s}_n)^{-1}$

3.1.2 映射

上一节已经得到了发送符号的估计值 \hat{x}_n , 对于传统线性均衡器来说, 其结果可以直接给译码器, 最后得到译码结果。但是, 由于采用的是 Turbo 均衡算法, 因此译码器采用软输入软输出算法, 因此, 需要关于发送符号的软信息。具体可以看图 3-2。

基于 MMSE 准则的线性 Turbo 均衡器计算其后验对数似然比 LLR 值:

$$\begin{aligned}L(c_{n,j}|\hat{x}_n) &\triangleq \ln \frac{P(c_{n,j} = 0|\hat{x}_n)}{P(c_{n,j} = 1|\hat{x}_n)} \\ &= \ln \frac{\sum_{\forall \mathbf{c}_n: c_{n,j}=0} p(\hat{x}_n|\mathbf{c}_n)P(\mathbf{c}_n)}{\sum_{\forall \mathbf{c}_n: c_{n,j}=1} p(\hat{x}_n|\mathbf{c}_n)P(\mathbf{c}_n)}\end{aligned}\quad (3-17)$$

上式可以分解如下:

$$\ln \underbrace{\frac{\sum_{\forall \mathbf{c}_n: c_{n,j}=0} p(\hat{x}_n|\mathbf{c}_n) \prod_{\forall j' \neq j} P(c_{n,j'})}{\sum_{\forall \mathbf{c}_n: c_{n,j}=1} p(\hat{x}_n|\mathbf{c}_n) \prod_{\forall j' \neq j} P(c_{n,j'})}}_{L_e^E(c_{n,j})} + L_a(c_{n,j}) \quad (3-18)$$

其中, $L_a(c_{n,j})$ 为上一次译码器反馈的外部信息 $L_e^D(c_{n,j})$, 作为均衡器的先验信息。 $L_e^E(c_{n,j})$ 是均衡器输出的软信息, 也是要通过 \hat{x}_n 映射得到的。

假设 PDF $p(\hat{x}_n | \mathbf{c}_n = \mathbf{s}_i) = p(\hat{x}_n | x_n = \alpha_i)$, $i = 1, 2, \dots, 2^Q$ 是均值为 $\mu_{n,i} \triangleq E(\hat{x}_n | x_n = \alpha_i)$, 方差为 $\sigma_{n,i}^2 = \text{Cov}(\hat{x}_n, \mathbf{x}_n | x_n = \alpha_i)$ 的高斯分布:

$$p(\hat{x}_n | \mathbf{c}_n = \mathbf{s}_i) \approx \phi_{\mu_{n,i}, \sigma_{n,i}^2}(\hat{x}_n) \quad (3-19)$$

这个假设可以简化 $L_e^E(c_{n,j})$ 的计算, 而且只应用于 \hat{x}_n 到 $L_e^E(c_{n,j})$ 的映射, 因此性能损失是可以忽略不计的。而且发现 $L_e^E(c_{n,j})$ 或 $L(c_{n,j} | \hat{x}_n)$ 对于小范围的性能损失的鲁棒性非常好。

\hat{x}_n 的统计量 $\mu_{n,i}$ 和 $\sigma_{n,i}^2$ 如下:

$$\begin{aligned} \mu_{n,j} &= \mathbf{f}_n^H (E(\mathbf{z}_n | x_n = \alpha_i) - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}_n) \\ &= \mathbf{f}_n^H (\mathbf{H}_n E(\mathbf{x}_n | x_n = \alpha_i) - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}_n) \\ &= \alpha_i \cdot \mathbf{f}_n^H \mathbf{s}_n \\ \sigma_{n,i}^2 &= \mathbf{f}_n^H \text{Cov}(\mathbf{z}_n, \mathbf{z}_n | x_n = \alpha_i) \mathbf{f}_n \\ &= \mathbf{f}_n^H (\Sigma_n - v_n \mathbf{s}_n \mathbf{s}_n^H) \mathbf{f}_n \\ &= \mathbf{f}_n^H \mathbf{s}_n - v_n \mathbf{f}_n^H \mathbf{s}_n \mathbf{s}_n^H \mathbf{f}_n \end{aligned} \quad (3-20)$$

从而得到:

$$\begin{aligned} L_e^E(c_{n,j}) &= \ln \frac{\sum_{\forall \mathbf{s}_i: s_{i,j}=0} p(\hat{x}_n | \mathbf{c}_n = \mathbf{s}_i) \prod_{\forall j': j' \neq j} P(c_{n,j'} = s_{i,j'})}{\sum_{\forall \mathbf{s}_i: s_{i,j}=1} p(\hat{x}_n | \mathbf{c}_n = \mathbf{s}_i) \prod_{\forall j': j' \neq j} P(c_{n,j'} = s_{i,j'})} \\ &= \ln \frac{\sum_{\forall \mathbf{s}_i: s_{i,j}=0} \exp \left(-\rho_{n,j} + \sum_{\forall j': j' \neq j} \tilde{s}_{i,j'} L(c_{n,j'}) / 2 \right)}{\sum_{\forall \mathbf{s}_i: s_{i,j}=1} \exp \left(-\rho_{n,j} + \sum_{\forall j': j' \neq j} \tilde{s}_{i,j'} L(c_{n,j'}) / 2 \right)} \end{aligned} \quad (3-21)$$

其中,

$$\begin{aligned} \rho_{n,j} &= \frac{|\bar{x}_n - \mu_{n,i}|^2}{\sigma_{n,i}^2} \\ &= \frac{|\mathbf{f}_n^H (\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}_n) - \alpha_i \mathbf{f}_n^H \mathbf{s}_n|^2}{\mathbf{f}_n^H \mathbf{s}_n - v_n \mathbf{f}_n^H \mathbf{s}_n \mathbf{s}_n^H \mathbf{f}_n} \end{aligned} \quad (3-22)$$

对于3-1的映射方式, 则外部信息 $L_e^E(c_{n,j})$ 可以简化为:

表 3-3 $L_e^E(c_{n,j})$ 的计算公式

QPSK
$L_e^E(c_{n,1}) = \sqrt{8}/(1 - v_n \mathbf{s}_n^H \mathbf{f}_n) \cdot \text{Re}(\mathbf{f}_n^H (\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}_n))$
$L_e^E(c_{n,2}) = \sqrt{8}/(1 - v_n \mathbf{s}_n^H \mathbf{f}_n) \cdot \text{Im}(\mathbf{f}_n^H (\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}_n))$

如果先验信息不存在,也即是 $L_e^D(c_{n,j}) = 0$ 对所有的 n, j , 那么 \mathbf{f}_n 是固定的, 不随时间而改变。在此条件下, $\bar{x}_n = 0, v_n = 1$, 那么 \mathbf{f}_n 可以改写如下:

$$\mathbf{f}_{NA} \triangleq \Sigma_n^{-1}|_{v_n=1, \forall n} = (\sigma_\omega^2 \mathbf{I}_N + \mathbf{H}_n \mathbf{H}_n^H)^{-1} \mathbf{s}_n \quad (3-23)$$

此式与传统的 MMSE 均衡器一样, NA 代表的是“没有先验信息”。对于传统线性均衡器, 可以采用最陡下降法来降低运算复杂度, 然而本文采用的基于先验信息 MMSE 准则的线性 Turbo 均衡算法中, \mathbf{f}_n 依赖于 \mathbf{H}_n 和 \mathbf{V}_n , 最陡下降法不能适用, 因此必须计算矩阵的逆, 直接实现此操作需要 N^3 的计算量。

3.1.3 算法优化与总结

分析 Σ_n 的结构从而发展出一个快速递归求解 \mathbf{f}_n 的算法, 此算法只需要 N^2 的计算量。很多关于自适应滤波器的文献^[52,75] 都有相关的快速算法。本文主要参考文献[33]中的快速算法。时间递归更新算法如下:

$$\Sigma_n \triangleq \begin{bmatrix} \sigma_P & \sigma_P^H \\ \sigma_P & \Sigma_P \end{bmatrix}, \quad \Sigma_{n+1} \triangleq \begin{bmatrix} \Sigma_N & \sigma_N \\ \sigma_N^H & \sigma_N \end{bmatrix} \quad (3-24)$$

其中, $\Sigma_i, i \in \{P, N\}$ 是 $(N-1) \times (N-1)$ 的矩阵, σ_i 是长度为 $N-1$ 的列向量, σ_i 则是标量。下标 P 代表当前“Present”时间 n 。而 N 代表下一“next”时间 $n+1$ 。相似的, 对于 Σ_n, Σ_{n+1} 的逆的划分如下:

$$\begin{aligned} \Sigma_n^{-1} &\triangleq \mathbf{U}_n \triangleq \begin{bmatrix} u_P & \mathbf{u}_P^H \\ \mathbf{u}_P & \mathbf{U}_P \end{bmatrix} \\ \Sigma_{n+1}^{-1} &\triangleq \mathbf{U}_{n+1} \triangleq \begin{bmatrix} \mathbf{U}_N & \mathbf{u}_N \\ \mathbf{u}_N^H & u_N \end{bmatrix} \end{aligned} \quad (3-25)$$

其中,用到了 Hermitian 矩阵的逆依然是 Hermitian 矩阵这一性质。从上式可以发现, Σ_P 和 Σ_N 是一致的:

$$\Sigma_P = \Sigma_N = \sigma_\omega^2 \mathbf{I}_N + \mathbf{H}'_n \text{diag}[v_{n-M+N_2+2}, \dots, v_{n+N_1}] \mathbf{H}_n^H \quad (3-26)$$

其中, \mathbf{H}' 是 $(N-1) \times (N+M-2)$ 的信道卷积矩阵。基于这个事实,递归算法从 \mathbf{U}_n 计算 Σ_n^{-1} ,并令 $\Sigma_P^{-1} = \Sigma_N^{-1}$,进而从 Σ_N^{-1} 计算得到 \mathbf{U}_{n+1} 。

Σ_n 的子矩阵 Σ_P 的逆 Σ_P^{-1} 可以通过解 $\Sigma_n \mathbf{U}_n = \mathbf{I}_N$ 方程得到:

$$\begin{aligned} \Sigma_P \mathbf{U}_P + \sigma_P \mathbf{u}_P^H &= \mathbf{I}_{N-1} \\ \Sigma_P \mathbf{u}_P + \sigma_P u_P^H &= \mathbf{0}_{N-1} \\ \rightarrow \Sigma_P^{-1} &= \mathbf{U}_P - \mathbf{u}_P \mathbf{u}_P^H / u_P \end{aligned} \quad (3-27)$$

通过求解方程 $\Sigma_{n+1} \mathbf{U}_{n+1} = \mathbf{I}_N$,可以用 Σ_N, σ_N 以及 σ_N 来表示 \mathbf{U}_N 和 \mathbf{u}_N :

$$\begin{aligned} \sigma_N'^{-1} &\triangleq \Sigma_N^{-1} \sigma_N \\ u_N &= 1/(\sigma_N - \sigma_N^H \sigma_N') \\ \mathbf{u}_N &= -u_N \sigma_N' \\ \mathbf{U}_N &= \Sigma_N^{-1} + u_N \sigma_N' \sigma_N'^H \end{aligned} \quad (3-28)$$

其中,利用中间向量 σ_N' 来简化算法。矩阵 Σ_N^{-1} 和矩阵 Σ_P^{-1} 是相等的,而且 σ_N 和 σ_N 可以利用式3-12和3-27得到:

$$\begin{bmatrix} \sigma_N \\ \sigma_N \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{N-1} \\ \sigma_\omega^2 \end{bmatrix} + \mathbf{H}_n \mathbf{V}_{n-1} \mathbf{H}_n^H \begin{bmatrix} \mathbf{0}_{N-1} \\ 1 \end{bmatrix} \quad (3-29)$$

此时,可以利用 $\mathbf{U}_N, \mathbf{u}_N$ 和 u_N 计算出 \mathbf{U}_n ,从而形成完整的递归算法。当 $\mathbf{f}_{n+1} = \Sigma_{n+1}^{-1} \mathbf{s}_{n+1}$ 得到之后,就可以利用式3-21求解出外部信息 $L_e^E(c_{n+1,j})$ 。

为了利用这种时间递归算法, \mathbf{U}_n 在 $n=1$ 的初始化值 $\mathbf{U}_1 = (\sigma_\omega \mathbf{I}_N + \mathbf{H}_1 \mathbf{V}_1 \mathbf{H}_1^H)^{-1}$ 需要给出。由于算法在训练序列的时候就开始递归计算,因此 $\mathbf{V}_1 = 0$,从而得到 $\mathbf{U}_1 = \sigma_\omega^{-2} \mathbf{I}_N$ 。算法步骤总结如表3-4

表 3-4 MMSE-LE 递归算法总结

输入:

- 星座图映射 $\mathcal{S} = \alpha_1, \dots, \alpha_{2^Q}$,
- 滤波器长度 N_1 和 N_2 ,
- 信道特征 $h_k, k = 0, \dots, M-1$ 和 σ_ω^2 ,
- 接收符号 $z_n, n = 1 - N_2, \dots, L + N_1$,
- 先验信息 $L_e^D(c_{n,j}), n = 1 - N_2 + M + 1, \dots, L + N_1, j = 1, \dots, Q$,

初始化:

- 定义变量 $\mathbf{f} = \mathbf{0}_N, \tilde{\mathbf{U}} = \mathbf{0}_{N \times (N-1)}$,
 $\mathbf{u} = \mathbf{u}' = \mathbf{0}_{N-1}, x = u = \mu = \rho_i = 0, i = 1, \dots, 2^Q$,
- 计算 \bar{x}_n 和 $v_n, n = 1 - N_2 - M + 1, \dots, L + N_1$,
- 计算 $\bar{z}_n, n = 1 - N_2, \dots, L + N_1$,
- 计算 $\tilde{\mathbf{U}} = (\sigma_\omega^2 \mathbf{I}_N + \mathbf{H}_1 \mathbf{V}_1 \mathbf{H}_1^H)^{-1}$,

续下页

续表 3-4 MMSE-LE 递归算法总结

均衡算法:

FOR $n = 1$ TO L DO

$$\mathbf{f} = \tilde{\mathbf{U}}\mathbf{s},$$

$$\mu = \mathbf{f}^H \mathbf{s},$$

$$x = \mathbf{f}^H (\mathbf{z}_n - \bar{\mathbf{z}}_n) + \bar{x}_n \mu,$$

FOR $i = 1$ TO 2^Q DO

$$\rho_i = |x - \alpha_i \mu|^2 / (\mu - \mu^2),$$

END

FOR $j = 1$ TO Q DO

$$L_e^E(c_{n,j}) = \ln \frac{\sum_{\forall \mathbf{s}_i: s_{i,j}=0} \exp \left(-\rho_{n,j} + \sum_{\forall j': j' \neq j} \tilde{s}_{i,j'} L(c_{n,j'}) / 2 \right)}{\sum_{\forall \mathbf{s}_i: s_{i,j}=1} \exp \left(-\rho_{n,j} + \sum_{\forall j': j' \neq j} \tilde{s}_{i,j'} L(c_{n,j'}) / 2 \right)}$$

END

IF $n < L$ THEN

$$\begin{bmatrix} \mathbf{U} & \mathbf{u} \\ \mathbf{u}^N & u \end{bmatrix} = \tilde{\mathbf{U}},$$

$$\mathbf{U} = \mathbf{U} - \mathbf{u}\mathbf{u}^H / u,$$

$$\begin{bmatrix} \mathbf{u} & u \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{N-1} \\ \sigma_\omega^2 \end{bmatrix} = \mathbf{H}_{n+1} \mathbf{V}_{n+1} \mathbf{H}_{n+1}^H,$$

$$\mathbf{u}' = \mathbf{U}\mathbf{u},$$

$$u = 1 / (u - \mathbf{u}^H \mathbf{u}'),$$

$$\mathbf{u} = -u\mathbf{u}',$$

$$\mathbf{U} = \mathbf{U} + u\mathbf{u}'\mathbf{u}'^H,$$

$$\tilde{\mathbf{U}} = \begin{bmatrix} u & \mathbf{u}^H \\ \mathbf{u} & \mathbf{U} \end{bmatrix},$$

END

END

3.2 低复杂度近似算法

为了进一步减少计算量,寻找不随 n 变化的滤波器系数 $\mathbf{f} \triangleq [f_{N_2}^*, f_{N_2-1}^*, \dots, f_{-N_1}^*]$, 提出 MMSE 均衡算法的低复杂度近似解决方案来代替式??获取 \hat{x}_n 。其中,利用 $\Sigma_n + (1 - v_n)\mathbf{ss}^H$ 的时间平均值来代替变化值:

$$\begin{aligned}\mathbf{f}' &\triangleq \left(\frac{1}{L} \sum_{n=1}^L \Sigma_n + (1 - \bar{v})\mathbf{ss}^H \right) \\ &= (\sigma_\omega^2 \mathbf{I}_N + \mathbf{H}\bar{\mathbf{V}}\mathbf{H}^H \gamma \gamma + (1 - \bar{v})\mathbf{ss}^H)^{-1} \mathbf{s}\end{aligned}\quad (3-30)$$

其中 $\bar{\mathbf{V}} \triangleq \frac{1}{L} \sum_{n=1}^L \mathbf{V}_n$ 和 $\bar{v} = \frac{1}{L} = \sum_{n=1}^L v_n$ 。发送符号估计值 \hat{x}_n 可以改写成:

$$\hat{x}_n = \mathbf{f}'^H (\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}) \quad (3-31)$$

定义向量:

$$\mathbf{f} \triangleq (\sigma_\omega^2 \mathbf{I}_N + \mathbf{H}\bar{\mathbf{V}}\mathbf{H}^H)^{-1} \mathbf{s} \quad (3-32)$$

与上面算法相类似, \mathbf{f}' 也是 \mathbf{f} 的伸展:

$$\mathbf{f}' = (1 + (1 - \bar{v})\mathbf{f}^H \mathbf{f})^{-1} \mathbf{f} \quad (3-33)$$

\hat{x}_n 的均值和方差可以计算如下:

$$\begin{aligned}\mu_{n,i} &= K \cdot \mathbf{f}'^H (\mathbf{E}(\mathbf{z}_n | x_n = \alpha_i) - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}) \\ &= K \cdot \alpha_i \cdot \mathbf{f}'^H \mathbf{s} \\ \sigma_{n,i}^2 &= K^2 \cdot \mathbf{f}'^H \text{Cov}(\mathbf{z}_n, \mathbf{z}_n | x_n = \alpha_i) \mathbf{f}' \\ &= K^2 \cdot \mathbf{f}'^H (\Sigma_n - v_n \mathbf{ss}^H) \mathbf{f}'\end{aligned}\quad (3-34)$$

其中 $K \triangleq (1 + (1 - \bar{v})\mathbf{s}^H \mathbf{f})^{-1}$, 从而得到:

$$\rho_{n,i} = \frac{|\mathbf{f}'^H (\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}) - \alpha_i \mathbf{f}'^H \mathbf{s}|^2}{\mathbf{f}'^H (\sigma_\omega^2 \mathbf{I}_H + \mathbf{H}\bar{\mathbf{V}}_n \mathbf{H}^H - v_n \mathbf{ss}^H) \mathbf{f}'} \quad (3-35)$$

利用式3-17可以求出均衡器输出的外部软信息 $L_e^E(c_{n,j})$ 。

对于 L 很长的情况下, $\bar{\mathbf{V}}$ 可以用 $\bar{v} \mathbf{I}_{N+M-1}$ 代替, 而此中替换对于 SISO 均衡器的性能影响很小, 因此可以简化 \mathbf{f} 为:

$$\hat{\mathbf{f}} \triangleq (\sigma_\omega^2 \mathbf{I}_N + \bar{v} \mathbf{H} \mathbf{H}^H)^{-1} \mathbf{s} \quad (3-36)$$

$\rho_{n,i}$ 可以简化如下:

$$\begin{aligned}
 & \frac{|\mathbf{f}^H(\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}) - \alpha_i \mathbf{f}^H \mathbf{s}|^2}{\mathbf{f}^H(\sigma_\omega^2 \mathbf{I}_N + \mathbf{H} \mathbf{V}_n \mathbf{H}^H - v_n \mathbf{s} \mathbf{s}^H) \mathbf{f}} \\
 & \approx \frac{|\hat{\mathbf{f}}^H(\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}) - \alpha_i \hat{\mathbf{f}}^H \mathbf{s}|^2}{\hat{\mathbf{f}}^H(\sigma_\omega^2 \mathbf{I}_N + \bar{v}(\mathbf{H} \mathbf{H}^H - \mathbf{s} \mathbf{s}^H)) \hat{\mathbf{f}}} \\
 & = \frac{|\hat{\mathbf{f}}^H(\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}) - \alpha_i \hat{\mathbf{f}}^H \mathbf{s}|^2}{\hat{\mathbf{f}}^H \mathbf{s} (1 - \mathbf{s}^H \hat{\mathbf{f}})}
 \end{aligned} \tag{3-37}$$

对于 QPSK, 且映射方式按照表3-1, 那么外部信息可以简化为表3-5

表 3-5 $L_e^E(c_{n,j})$ 的简化计算公式

QPSK	
$\mu = \hat{\mathbf{f}}^H \mathbf{s}, \mathbf{p} = \mathbf{H}^H \hat{\mathbf{f}}, K_f = \hat{\mathbf{f}}^H \hat{\mathbf{f}}, x = \hat{\mathbf{f}}^H(\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n \mathbf{s}),$	
$L_e^E(c_{n,1}) = \sqrt{8} \cdot \mu \text{Re}(x) / (\sigma_\omega^2 K_f + \mathbf{p}^H \mathbf{V}_n \mathbf{p} - v_n \mu ^2),$	
$L_e^E(c_{n,2}) = \sqrt{8} \cdot \mu \text{Im}(x) / (\sigma_\omega^2 K_f + \mathbf{p}^H \mathbf{V}_n \mathbf{p} - v_n \mu ^2),$	

3.3 仿真与分析

下面从几个方面对上面介绍的 Turbo 均衡算法进行仿真分析:

3.3.1 不同均衡算法之间的比较

表 3-6 不同均衡算法比较的参数设置

参数项		参数值
信道类型	ch_none	无码间干扰
	CHA	[0.407 0.815 0.407]
均衡器类型	eq_map_det	MAP 均衡算法
	eq_exact_lin	MMSE 精确线性均衡算法
	eq_approx_lin	MMSE 近似线性均衡器
调制方式	mo_bpsk	BPSK
交织长度	bl_1024	1024

续下页

续表 3-6 不同均衡算法比较的参数设置

编码长度	co_r2_m2	[5 7]
删余方式	pu_1	无删余
编码类型	RSC	递归系统卷积码

参数如表3-6,仿真结果如图3-3 可以从两方面分析图3-3,

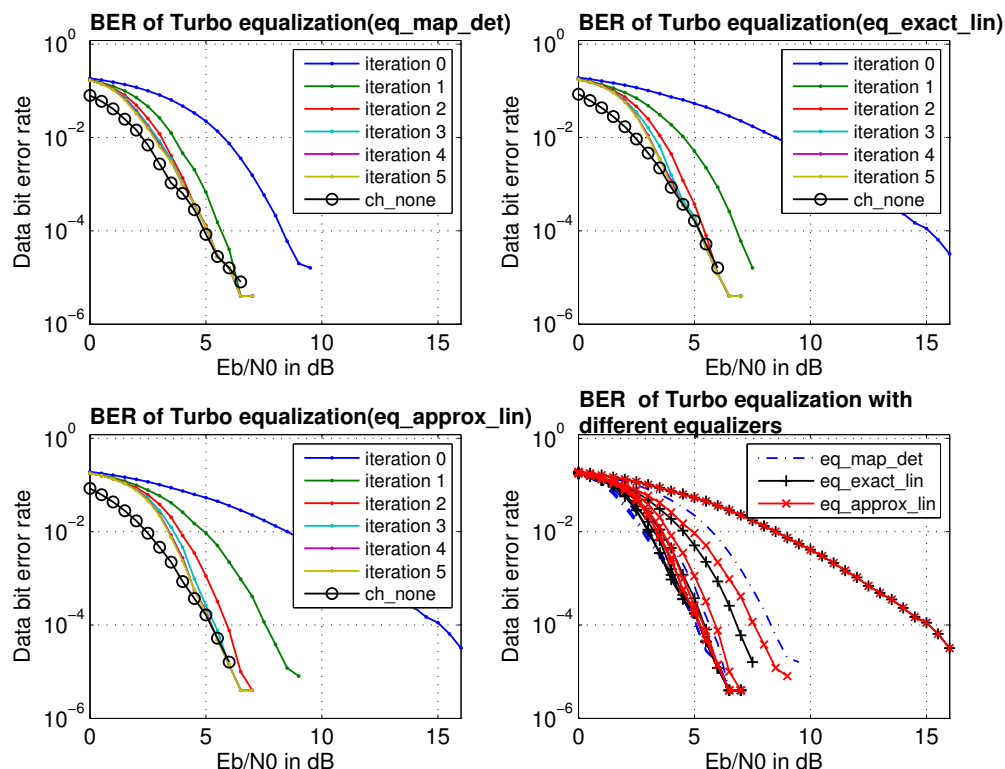


图 3-3 不同均衡算法的性能比较

1. 单独分析每一个图,可以看出,随着迭代次数的增加,所有均衡算法的误码率曲线都越来越接近 AWGN 下的误码率曲线。这说明,随着迭代次数的增加,外部信息被利用的越来越充分。
2. 分析图3-3中最后一个小图,可以看出,MAP 均衡算法是最好的,但是随着迭代次数的增加,MMSE 算法以及其近似算法越来越接近 MAP 均衡性能。因

此,用基于先验信息 MMSE 准则的线性 Turbo 均衡算法代替 MAP 算法对性能的损失很小。

3.3.2 不同调制方式之间的比较

表 3-7 不同调制方式均衡性能比较的参数设置

参数项		参数值
信道类型	ch_none	无码间干扰
	CHA	[0.407 0.815 0.407]
均衡类型	eq_exact_lin	MMSE 精确线性均衡算法
调制方式	mo_bpsk	BPSK
	mo_qpsk	QPSK
	mo_8psk	8PSK
交织长度	bl_1024	1024
编码长度	co_r2_m2	[5 7]
删余方式	pu_1	无删余
编码类型	RSC	递归系统卷积码

参数如表3-7,仿真结果如图3-4 可以从两方面分析图3-4,

1. 单独分析每一个图,可以看出,随着迭代次数的增加,所有调制方式下的误码率曲线都越来越接近 AWGN 下的误码率曲线。这说明,随着迭代次数的增加,外部信息被利用的越来越充分。
2. 分析图3-4中最后一个小图,可以看出,BPSK 的性能最好,QPSK 的性能次之,最差的是 8PSK,究其原因,8PSK 调制下的码字之间的欧式距离最小,最容易出错,因此性能最差,但是 8PSK 的码率最高,在实际应用中要综合考虑性能和码率之间的平衡。本文所在的水声相干通信系统中综合考虑这两方面,采用的是 QPSK 调制方式。

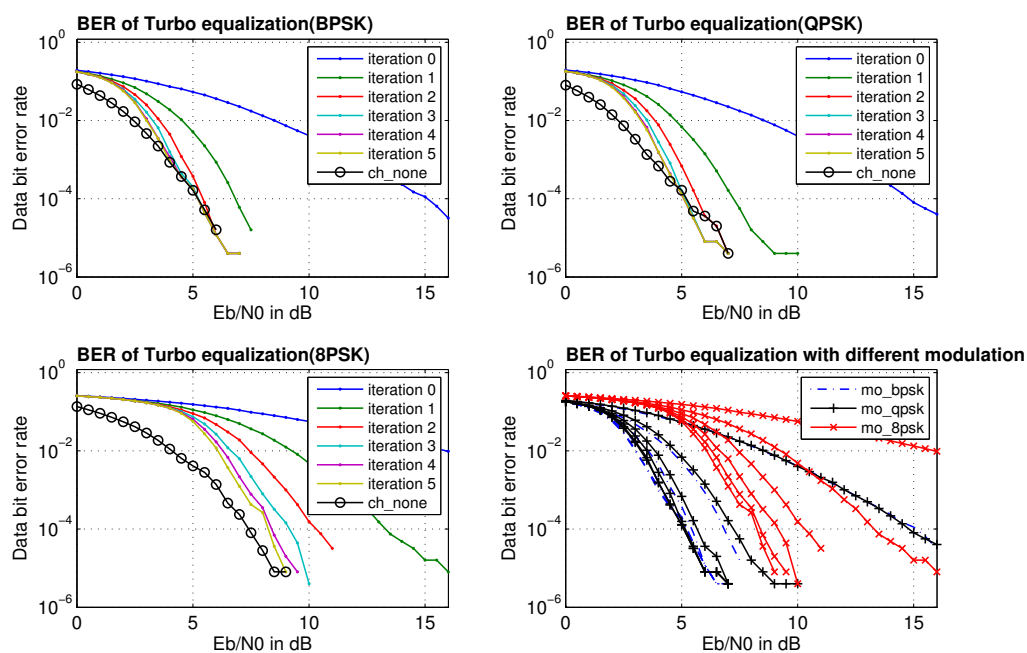


图 3-4 不同调制方式下均衡性能的比较

3.3.3 不同编码方式之间的比较

表 3-8 不同编码方式均衡性能比较的参数设置

参数项		参数值
信道类型	ch_none	无码间干扰
	CHA	[0.407 0.815 0.407]
均衡类型	eq_exact_lin	MMSE 精确线性均衡算法
调制方式	mo_bpsk	BPSK
交织长度	bl_1024	1024
编码方式	co_r2_m2	[5 7]
	co_r2_m4	[19 29]
删余方式	pu_1	无删余

续下页

续表 3-8 不同编码方式均衡性能比较的参数设置

编码类型	RSC	递归系统卷积码
------	-----	---------

参数如表3-8,仿真结果如图3-5 可以从两方面分析图3-5,

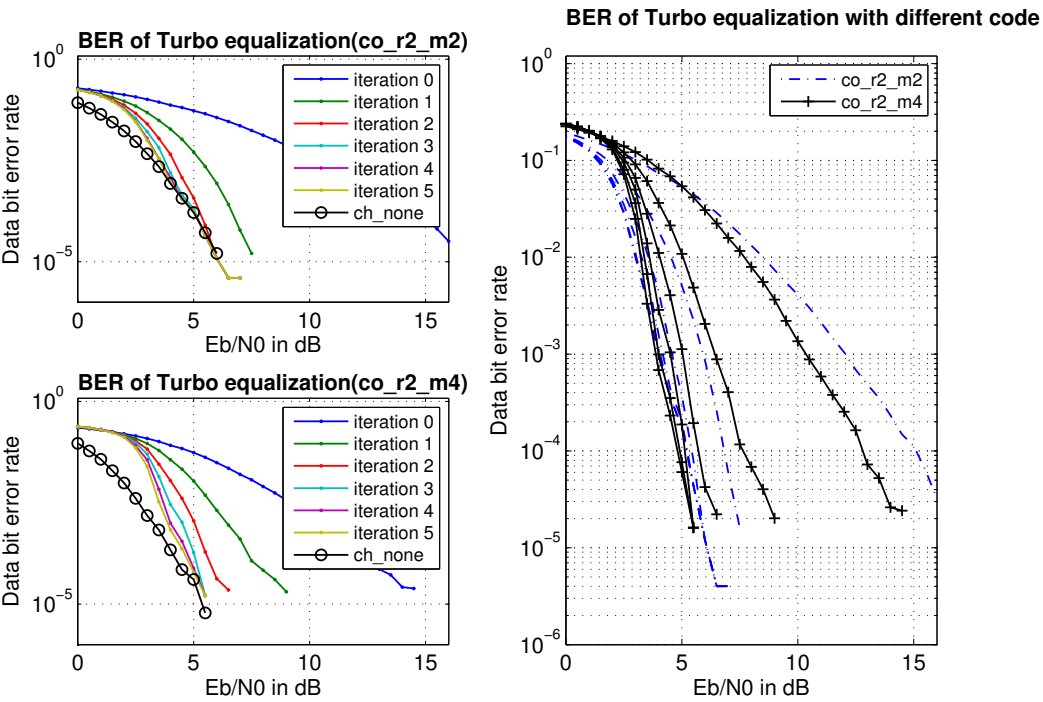


图 3-5 不同调制方式下均衡性能的比较

1. 单独分析每一个图,可以看出,随着迭代次数的增加,所有编码方式下的误码率曲线都越来越接近 AWGN 下的误码率曲线。这说明,随着迭代次数的增加,外部信息被利用的越来越充分。
2. 分析图3-5中最后一个小图,可以看出,约束长度越大的编码方式,其性能越好 (每一次迭代的比较结果,都是约束长度大的编码方式更优)。因为约束长度越大的编码方式,其译码性能越好,反馈给均衡器的外部信息越准确,因此,总体的译码性能会越好。考虑到这个问题,本文在湖试数据处理部分采用的是纠错性能极强的 Turbo 码。

3.3.4 不同删余方式之间的比较

表 3-9 不同删余方式均衡性能比较的参数设置

参数项		参数值
信道类型	ch_none	无码间干扰
	CHA	[0.407 0.815 0.407]
均衡类型	eq_exact_lin	MMSE 精确线性均衡算法
调制方式	mo_bpsk	BPSK
交织长度	bl_1024	1024
编码方式	co_r2_m2	[5 7]
删余方式	pu_1	无删余
	pu_100	3/4
编码类型	RSC	递归系统卷积码

参数如表3-9,仿真结果如图3-6 可以从两方面分析图3-6,

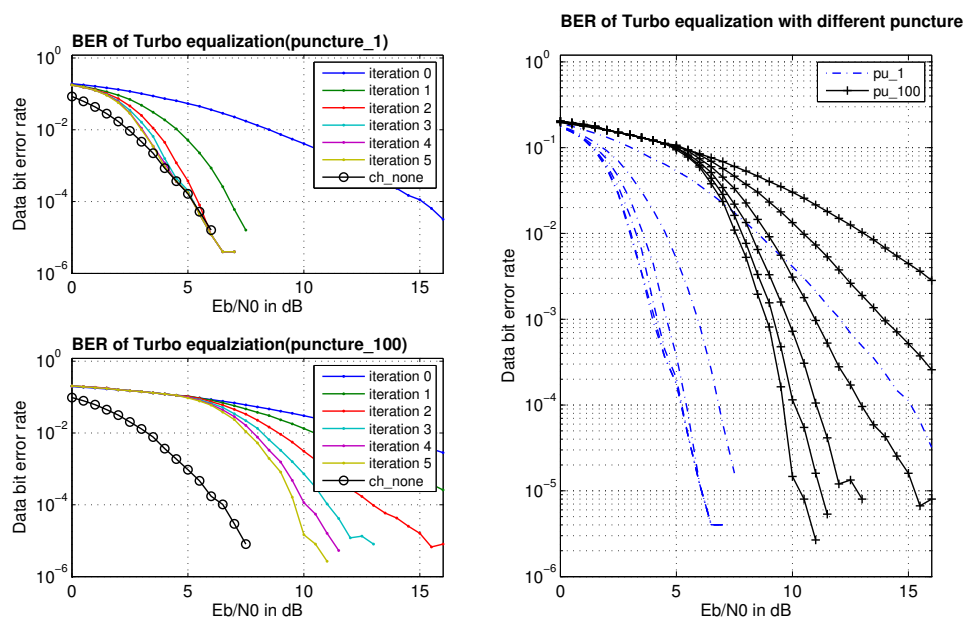


图 3-6 不同删余方式下均衡性能的比较

1. 单独分析每一个图,可以看出,随着迭代次数的增加,所有删余方式下的误码

率曲线都越来越接近 AWGN 下的误码率曲线。这说明,随着迭代次数的增加,外部信息被利用的越来越充分。

2. 分析图3-6中最后一个小图,可以看出,删余越小,其性能越好 (每一次迭代的比较结果,都是删余小的均衡性能更优)。因为删余越小,损失的信息越少,其译码性能越好,反馈给均衡器的外部信息越准确,因此,总体的译码性能会越好。但是删余越少,其码率越低,考虑到这个问题,本文在湖试数据处理部分采用的删余方式为 `pu_100`。

3.3.5 不同交织长度之间的比较

表 3-10 不同交织长度均衡性能比较的参数设置

参数项		参数值
信道类型	<code>ch_none</code>	无码间干扰
	<code>CHA</code>	[0.407 0.815 0.407]
均衡类型	<code>eq_exact_lin</code>	MMSE 精确线性均衡算法
调制方式	<code>mo_bpsk</code>	BPSK
交织长度	<code>bl_1024</code>	1024
	<code>bl_65535</code>	65535
编码方式	<code>co_r2_m2</code>	[5 7]
删余方式	<code>pu_1</code>	无删余
编码类型	<code>RSC</code>	递归系统卷积码

参数如表3-10,仿真结果如图3-7 可以从两方面分析图3-7,

1. 单独分析每一个图,可以看出,随着迭代次数的增加,所有交织长度下的误码率曲线都越来越接近 AWGN 下的误码率曲线。这说明,随着迭代次数的增加,外部信息被利用的越来越充分。
2. 分析图3-7中最后一个小图,两种交织长度下的均衡性能基本一致,这是因为,在交织长度很小的情况下,随着交织长度的增加,均衡和译码的性能都会

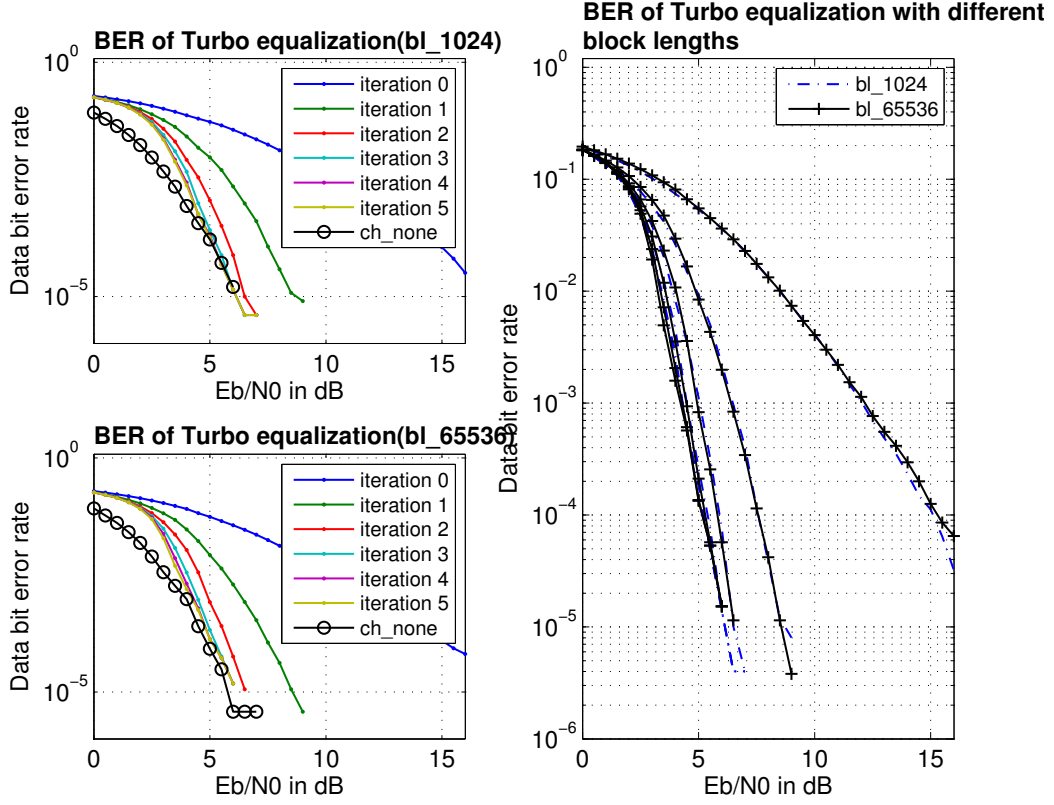


图 3-7 不同调制方式下均衡性能的比较

有所提升,但是当交织长度增大到一定值之后,再增加交织长度并不会提高均衡和译码性能,而相反会增加均衡和译码的计算量。综合考虑这个问题,本文在湖试数据处理部分采用的交织长度为 1936。

3.4 分数间隔线性 SISO 均衡算法

上面介绍的都是采用符号采样率的均衡器,这在多普勒比较严重的水声信道中容易产生相位翻转的现象,为了解决这个问题,通常采用分数间隔均衡器,下面将介绍采样率为二倍符号率的线性 SISO 均衡算法。

当接收信号采用二倍符号速率采样时,接收符号可以改写为:

$$z_{n,i} = \sum_{l=0}^{M-1} h_{n,l,i} x_{n-l} + \omega_{n,i}, \quad i \in \{0, 1\} \quad (3-38)$$

其中 $\mathbf{h}_{n,i} = [h_{n,0,i}, \dots, h_{n,M-1,i}]^T$ 为分数间隔时变信道冲击响应。依然假设 N 为符号速率均衡器长度,那么真实的均衡器(分数间隔均衡器)长度为 $2N$,因此,为

了计算此时的发送符号的估计值 \hat{x}_n , 需要知道 $2N$ 个长度的接收符号 $z_{n,i}$ 以及 $N + M - 1$ 个长度的关于发送符号 x_n 的先验信息。重新定义向量 \mathbf{z}_n 和 $\boldsymbol{\omega}_n$, 使其长度为 $2N$ 。

$$\begin{aligned}\mathbf{z}_n &= [z_{n+N_1,1}, z_{n+N_1,0}, z_{n+N_1-1,1}, \dots, z_{n-N_2,1}, z_{n-N_2,0}]^T \\ \boldsymbol{\omega}_n &= [\omega_{n+N_1,1}, \omega_{n+N_1,0}, \omega_{n+N_1-1,1}, \dots, \omega_{n-N_2,1}, \omega_{n-N_2,0}]^T\end{aligned}\quad (3-39)$$

此时的信道卷积矩阵的大小为 $2N \times (N + M - 1)$:

$$\begin{aligned}\mathbf{H}_n &= \begin{bmatrix} \mathbf{h}_{n+N_1,1}^T & 0 & \dots & 0 \\ \mathbf{h}_{n+N_1,0}^T & 0 & \dots & 0 \\ 0 & \mathbf{h}_{n+N_1,1}^T & \dots & 0 \\ 0 & \mathbf{h}_{n+N_1,0}^T & \dots & 0 \\ & \dots & & \\ 0 & & \dots & \mathbf{h}_{n+N_1,1}^T \\ 0 & & \dots & \mathbf{h}_{n+N_1,0}^T \end{bmatrix} \\ &= \begin{bmatrix} h_{n+N_1,0,1} & h_{n+N_1,1,1} & \dots & h_{n+N_1,M-1,1} & 0 & \dots & 0 \\ h_{n+N_1,0,0} & h_{n+N_1,1,0} & \dots & h_{n+N_1,M-1,0} & 0 & \dots & 0 \\ 0 & h_{n+N_1-1,0,1} & \dots & h_{n+N_1-1,M-1,1} & 0 & \vdots & \\ 0 & h_{n+N_1-1,0,0} & \dots & h_{n+N_1-1,M-1,0} & 0 & \vdots & \\ \vdots & & \ddots & & 0 & & \\ 0 & \dots & 0 & h_{n-N_2,0,1} & h_{n-N_2,1,1} & \dots & h_{n-N_2,M-1,1} \\ 0 & \dots & 0 & h_{n-N_2,0,0} & h_{n-N_2,1,0} & \dots & h_{n-N_2,M-1,0} \end{bmatrix}\end{aligned}\quad (3-40)$$

假设噪声的方差 $\sigma_{\omega,n,i}^2 = E(\omega_{n,i}, \omega_{n,i}^*)$ 在每一个符号都是不同的, 而且是独立的, 因此噪声的协方差矩阵 $\boldsymbol{\Omega}_n = \text{Cov}(\boldsymbol{\omega}_n, \boldsymbol{\omega}_n)$ 为:

$$\boldsymbol{\Omega}_n = \text{diag}[\sigma_{\omega,n+N_1,1}^2, \sigma_{\omega,n+N_1,0}^2, \dots, \sigma_{\omega,n-N_2,1}^2, \sigma_{\omega,n-N_2,0}^2] \quad (3-41)$$

有了以上的改写, 接收符号序列为:

$$\mathbf{z}_n = \mathbf{H}_n \mathbf{x}_n + \boldsymbol{\omega} \quad (3-42)$$

此时, 接收符号的表达式与非分数间隔 SISO 均衡器的接收符号表达式是一致的, 因此, 可以利用非分数间隔均衡器的算法来求解分数间隔均衡器, 只是矩阵的维数为 $2N \times 2N$ 而不是原来的 $N \times N$ 。

为了避免计算矩阵的逆,依然采用类似与 3.1.1 节中的时间递归更新算法。定义 $\Sigma_n = \mathbf{H}_n \mathbf{V}_n \mathbf{H}_n^H + \Omega_n$ 和 $\mathbf{U}_n = \Sigma_n^{-1}$, 其维度为 $2N \times 2N$ 。与 3.1.1 节不同,此时对 Σ_n 和 \mathbf{U}_n 划分的四个部分都是矢量,为了方便区分,这里采用与 3.1.1 节不同的符号表示:

$$\begin{aligned} \Sigma_{n-1} &= \begin{bmatrix} \mathbf{A}_n & \mathbf{G}_n \\ \mathbf{G}_n^H & \mathbf{B}_n \end{bmatrix}, \quad \Sigma_n = \begin{bmatrix} \mathbf{E}_n & \mathbf{F}_n^H \\ \mathbf{F}_n & \mathbf{A}_n \end{bmatrix} \\ \mathbf{U}_{n-1} &= \begin{bmatrix} \mathbf{D}_n & \mathbf{N}_n \\ \mathbf{N}_n^H & \mathbf{M}_n \end{bmatrix}, \quad \mathbf{U}_n = \begin{bmatrix} \mathbf{K}_n & \mathbf{L}_n^H \\ \mathbf{L}_n & \mathbf{C}_n \end{bmatrix} \end{aligned} \quad (3-43)$$

其中, $\mathbf{B}_n, \mathbf{E}_n$ 和 \mathbf{M}_n 都是 2×2 的矩阵, $\mathbf{F}_n, \mathbf{G}_n, \mathbf{L}_n$ 和 \mathbf{N}_n 都是 $(2N-2) \times 2$ 的矩阵, $\mathbf{A}_n, \mathbf{C}_n$ 和 \mathbf{D}_n 都是 $(2N-2) \times (2N-2)$ 的矩阵。整体思路与 3.1.1 节一样,通过求解 $\Sigma_n \mathbf{U}_n = \Sigma_{n-1} \mathbf{U}_{n-1} = \mathbf{I}_{2N}$, 也就是

$$\begin{aligned} \mathbf{A}_n \mathbf{D}_n + \mathbf{G}_n \mathbf{N}_n^H &= \mathbf{I}_{2N-2} \\ \mathbf{A}_n \mathbf{D}_n + \mathbf{G}_n \mathbf{M}_n &= \mathbf{0}_{2N-2} \\ \mathbf{E}_n \mathbf{K}_n + \mathbf{F}_n^H \mathbf{L}_n &= \mathbf{I}_2 \\ \mathbf{F}_n \mathbf{K}_n + \mathbf{A}_n \mathbf{L}_n &= \mathbf{0}_{2N-2} \\ \mathbf{F}_n \mathbf{L}_n^H + \mathbf{A}_n \mathbf{C}_n &= \mathbf{I}_{2N-2} \end{aligned} \quad (3-44)$$

可以得到 $\mathbf{C}_n, \mathbf{K}_n$ 和 \mathbf{L}_n , 然后结合 $\mathbf{U}_{n-1}, \mathbf{E}_n$ 和 \mathbf{F}_n 可以得到 \mathbf{U}_n :

$$\begin{aligned} \mathbf{A}_n^{-1} &= \mathbf{D}_n - \mathbf{N}_n \mathbf{M}_n^{-1} \mathbf{N}_n^H \\ \Psi_n &= \mathbf{A}_n^{-1} \mathbf{F}_n \\ \mathbf{K}_n &= (\mathbf{E}_n - \mathbf{F}_n^H \Psi_n)^{-1} \\ \mathbf{L}_n &= -\Psi_n \mathbf{K}_n \\ \mathbf{C}_n &= \mathbf{A}_n^{-1} + \Psi_n \mathbf{K}_n \Psi_n^H \end{aligned} \quad (3-45)$$

此算法与上文中的符号间隔更新算法是一致的,当利用式 3-45 求解出 \mathbf{U}_n 之后,就可以用同样的方法求解 \mathbf{f}_n 和 \mathbf{f}_n^* 。

需要说明一下,本文在仿真的时候并没有实现分数间隔 MMSE 均衡算法,而是在湖试数据处理部分直接应用此算法。

3.5 Turbo 均衡与 Turbo 联合迭代

从上面的仿真分析可以看出,不论采用何种均衡算法 (MAP,MMSE-optimal,MMSE-LC) 都不能达到 AWGN 信道下的性能。而上文中的仿真采用的是卷积码,因此为了获得更好的均衡性能,可以采用纠错性能极好的 Turbo 码来代替卷积码。

当 turbo 均衡中引用 turbo 码作为系统的差错控制编码时,发送端需要两个交织器,一个交织器是用于两个分量编码器之间,另一个交织器是用于编码器与符号映射器/信道之间。在接收端,不考虑软迭代信道估计算法,总共有两处迭代回路,一处是两个 SISO 分量译码器之间的迭代,一处是 SISO 译码器与 SISO 均衡器的迭代。具体可以参考[28,35,36,38? ?]。

传统的结合 Turbo 码的 Turbo 均衡算法是在一次均衡迭代之中,Turbo 码本身进行了多次迭代,但文献[76]中提出一种新的关于均衡与译码之间迭代调度算法。在每一次均衡迭代中,只和 Turbo 译码器中的一个分量译码器进行迭代,而分量译码器之间交互信息是通过均衡器的迭代,相当于传统的算法是均衡器迭代为外部迭代,而译码器迭代为内部迭代,而文献[76]中将译码器迭代作为外部迭代而均衡器迭代作为内部迭代。此算法的优点是降低了复杂度。

3.6 采用预编码的线性 SISO 均衡算法

在前面介绍的迭代均衡和译码的主要思想是在均衡器和译码器之间通过传递软信息以迭代的方式进行联合译码检测。如果把卷积码和 ISI 信道看作是一个串行级联的卷积码 (SCCC)^[77],则卷积码就是外码,而把 ISI 信道看做是内码。在文[77]中显示,对于串行级联码仅作为内码是递归卷积码,级联码才有交织增益,而 ISI 信道是典型的有限脉冲响应 (FIR) 等效于一个非递归的卷积码,因此,不会获得交织增益,只能通过提高外码编码器的性能改善。

在本节研究通过采用二进制预编码技术,可以使得信道“表现”出递归特性,获得良好的交织增益。作为外码的卷积码编码器就可以使用小的约束长度实现高的编码增益,同时也降低了译码复杂度 (与卷积码约束长度相关)。

预编码的记忆长度要小于信道的记忆长度,这样通过二进制预编码后不改变网格状态数,因而并不增加线性均衡器复杂度,而最常用的是查分结构的预编码。

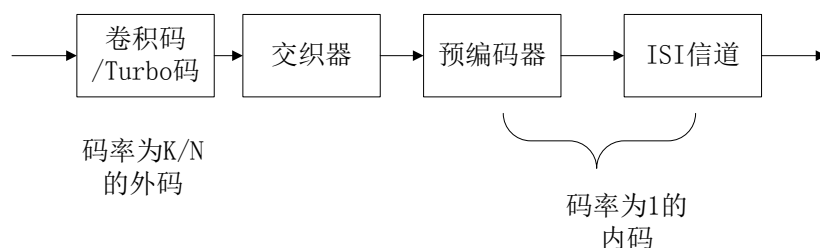


图 3-8 采用预编码的串行级联模型

文献[78]指出,采用预编码后,在低信噪比下,比没采用预编码的线性均衡器性能下降,是由于差错传播的原因。但是随着信噪比的增加,从误码率的下降斜率可以看出,由于获得了交织增益,性能取得了非常明显的改善,甚至比 AWGN 下的性能有 $1 - 2\text{dB}$ 的增益。

但是考虑到水声通信系统的实际情况,需要保证低信噪比下的信息传输的有效性,而预编码方案在低信噪比下性能反而会更差一些,不满足需求,综合考虑,本文在湖试数据处理的时候采用的 Turbo 码与 MMSE-LE 级联的均衡迭代方案。

3.7 本章小结

本章详细介绍了基于先验信息 MMSE 准则的线性 Turbo 均衡算法以及其近似算法,并为了避免矩阵求逆的操作,介绍了一种递归更新算法,使得运算复杂度大大减少。文中对介绍算法进行仿真,并分析该算法的优劣之处。

简单介绍了基于先验信息 MMSE 准则的线性 Turbo 均衡算法与 Turbo 码联合迭代,并给出几个参考。介于水声通信系统中相位变化的情况,介绍了该算法的分数间隔实现方式,用以解决相位翻转问题。

本章最后介绍了在发送端可以加入预编码来提高均衡性能,但是针对水声相干通信的特点,本文并没有采用这种方案。

第四章 软迭代信道估计算法

4.1 RS 码与水声通信

由于前面已经介绍了卷积码,已经在水声通信中使用卷积码,而且理论和实际当中,同等复杂度运算量的卷积码要比 RS 码的误码率性能要好,为什么这里还要研究 RS 码,并且作为一种信道编码方式应用到水声通信中?

在第三章中介绍了卷积码编码中几种码的终结方式,而且本设计也根据通信要求和实际情况选择了终结和咬尾两种码终结方式的结合,这就是意味着,编码的总长度比实际数据码字长度要多一个约束长度 K ,对于数据量很大的情况下,约束长度 K 就显得微不足道了,可以忽略不计,并不会导致多少带宽的浪费和码率的下降,而对于命令等几个字节甚至是几个比特数据量的时候,这就显得不合适了,对于序贯译码方式,通常选择约束长度 K 都很大,对于本设计仿真我们采用的是 $K = 24$,就是 3 个字节,可能远比要传输的数据量大,因此造成了很大的带宽浪费。因此采用 RS 码。

4.2 RS 码概述

RS¹码是由 MIT 林肯实验室的 Irving S. Reed 和 Gustave Solomon^[2]于 1960 年提出的一种基于多项式的纠错编码,Gorenstein 和 Zierler 于 1961 年证明了 RS 码与多进制 BCH 码的关系。Berlekamp 于 1967 年提出了直到至今还被广泛应用的硬判决译码算法,两年后,Massey 指出了此算法和线性反馈移位寄存器的关系,因此,该硬判决译码算法被称为 BM 算法,一直沿用至今。

RS 码具有很好的纠错特性,它是目前唯一实用的最小距离可分²码,能够有效地纠正随即符号错误,突发错误和删除错误。因此,在诞生初期,便被迅速应用。

¹Reed-Solomon

²MDS

4.3 RS 码的编码

RS 码是一种常见的线性分组码,同时它也是循环码。所谓循环码指的是任意码字循环移位后仍然在码字空间内的一类线性分组码^[9]。

一个 (n, k) 循环码可以用 $n - k$ 次的生成多项式 $g(x)$ 定义,其码多项式 $(n - 1)$ 次 $c(x)$ 是 $g(x)$ 的倍式: $c(x) = f(x)g(x)$, $k - 1$ 次多项式 $f(x)$ 称为消息多项式,其 k 个系数就是 k 个消息码元。由第二章的知识可知,如果 $g(x)$ 是 $x^n - 1$ 的因子,由 $g(x)$ 生成的码一定是循环码。

通过定义生成多项式,RS 码可以这样定义:在 $GF(q)$ 上, α 为本原域元素, $g(x)$ 定义为:

$$g(x) = (x - \alpha)(x - \alpha^2) \cdots (x - \alpha^{2t}) \quad (4-1)$$

其中 $2t = n - k, n = q - 1$

由第二章的知识可知,对 $GF(q)$ 而言,所有的 $q - 1$ 个非零元素就是 $x^{q-1} - 1$ 的所有根。那么,显然 $g(x)$ 是 $x^n - 1$ 的因子,因此,这样构造出来的码是循环码。

由4-1定义的 RS 码最小距离是 $2t + 1$,因此通过硬判决能纠正 t 个错误。RS 码的一个特点就是能够通过调整4-1根的数量很方便地调整其纠错能力。

对于能够纠错 t 个错误的 $RS(n, k, d)$ 码,具有如下特征:

1. **码长:** $n = 2^m - 1$ 符号或 $m(2^m - 1)$ 比特;
2. **信息码元数:** $k = n - 2t$ 符号或 mk 比特;
3. **监督码元数:** $n - k = 2t$ 符号或 $m(n - k)$ 比特;
4. **最小距离:** $d = 2t + 1 = n - k + 1$ 符号或 $m(n - k + 1)$ 比特

令信息码元多项式为:

$$m(x) = m_0 + m_1x + m_2x^2 + \cdots + m_{k-1}x^{k-1} \quad (4-2)$$

4.3.1 基于乘法形式的 RS 编码器

公式: $c(x) = m(x)g(x)$

结构图如下:图4-1具体实现步骤如下:

图 4-1 乘法编码器结构图

1. $2t$ 个寄存器全部清 0;
2. $m(x)$ 最高次系数 m_{k-1} 首先送入时,乘法器输出乘积的最高此项 x^{k+2t-1} 的系数 $m_{k-1}g_{2t}$,同时 m_{k-1} 存入寄存器的第一级;
3. $m(x)$ 的第二个系数 m_{k-2} 送入时, m_{k-1} 由第一级进入第二级寄存器,同时与 g_{2t-1} 相乘, $m_{k-2}g_{2t} + m_{k-1}g_{2t-1}$ 就是得到的乘积 x^{k+2t-2} 的系数;
4. 就这样重复进行,直到 $k + 2t$ 次移位后,乘法器输出乘积的常数项 m_0g_0

但是结构图4-1编码出来的 RS 码是非系统码,这并不是我们需要的,因此,下面将介绍系统码的编码器结构。

4.3.2 基于除法形式的 RS 编码器

设校验码多项式:

$$h(x) = h_k x^k + h_{k-1} x^{k-1} + \cdots + h_1 x + h + 0 \quad (4-3)$$

系统码的多项式:

$$C(x) = c_{n-1} x^{n-1} + c_{n-2} x^{n-2} + \cdots + c_{n-k} x^{n-k} + \cdots + c_1 x + c_0 \quad (4-4)$$

它的前 k 位系数: $c_{n-1}, c_{n-2}, \cdots, c_{n-k}$ 是已知的信息位,而后 $n - k$ 位系数: $c_{n-k-1}, c_{n-k-2}, \cdots, c_1, c_0$ 是要求的校验位。码多项式必是生成多项式 $g(x)$ 的倍式,所以

$$C(x) = m(x)g(x) \quad (4-5)$$

其中 $\partial C(x) \leq n-1, \partial g(x) = n-k, \partial m(x) \leq k-1$ 而

$$h(x)C(x) = m(x)g(x)h(x) = m(x)(x^n - 1) = m(x)x^n - m(x) \quad (4-6)$$

由于中 $\partial C(x) \leq n-1, \partial g(x) = n-k, \partial m(x) \leq k-1$, 所以 $m(x)x^n$ 的最低位次数至少为 n 次, 因而在 $h(x)C(x)$ 的乘积中 $x^{n-1}, x^{n-2}, \dots, x^k$ 的次数为 0。

而根据式4-6最左边的乘积可知, x^{n-1} 的系数:

$$c_{n-1-0}h_0 + c_{n-1-1}h_1 + \dots + c_{n-1-k}h_k \quad (4-7)$$

同理, x^{n-2} 的系数:

$$c_{n-2-0}h_0 + c_{n-2-1}h_1 + \dots + c_{n-2-k}h_k \quad (4-8)$$

根据上边的分析,

$$\sum_{j=0}^k c_{n-i-j}h_j = 0 \quad i = 0, 1, 2, \dots, n-k \quad (4-9)$$

由于 $h_k = 1$, 所以式4-9可以改写为:

$$c_{n-k-i} = -\sum_{j=0}^{k-1} c_{n-i-j}h_j \quad i = 1, 2, \dots, n-k \quad (4-10)$$

式4-10展开为:

$$\begin{aligned} c_{n-k-1} &= -(c_{n-1}h_0 + c_{n-2}h_1 + \dots + c_{n-k}h_{k-1}) \\ c_{n-k-2} &= -(c_{n-2}h_0 + c_{n-3}h_1 + \dots + c_{n-k-1}h_{k-1}) \\ &\vdots \\ c_{n-k-(n-k)} &= c_0 = -(c_k h_0 + c_{k-1}h_1 + \dots + c_1 h_{k-1}) \end{aligned} \quad (4-11)$$

由上式看出码字 C 的第一个码元 c_{n-k-1} 可由 k 个信息元 $c_{n-1}, c_{n-2}, \dots, c_{n-k}$ 与 $h(x)$ 的系数相乘得到, 而由 $c_{n-2}, c_{n-3}, \dots, c_{n-k}, c_{n-k-1}$ 可得到第二个校验元 c_{n-k-2} , 再由 c_{n-3}, \dots, c_{n-k} 信息元和第一、第二校验元 c_{n-k-1}, c_{n-k-2} 可得到第三校验元 c_{n-k-3} 。按这样的线性关系递推, 一直可求得所有的 $n-k$ 个校验元 $c_{n-k-1}, c_{n-k-2}, \dots, c_1, c_0$ 。因此, 基于校验多项式 $h(x)$ 构造除法编码器结构图如4-2 由于此种结构图编码出来的码字是系统码, 因此, 在本设计中, 采用的就是这种编码结构。

图 4-2 除法编码器结构图

4.4 RS 码的译码

就目前 RS 码的译码算法中,主要有硬判决算法和软判决译码算法,硬判决算法就是上面提到的 Berlekamp-Massey(BM) 算法,其纠错能力为 $t \leq (d_{\min} - 1)/2$, $d_{\min} = n - k + 1$ 为最小汉明距离。

而软判决译码算法有如下几种:

1. **由代数硬判决译码改进的准软判决译码算法³**: 由于删除信息可以看作最简单的软信息利用形式,Forney 提出的 GMD 译码算法通过删除最不可靠的码字,多次运行 BM 算法来实现译码,可以看作是 RS 码的准软译码算法; Chase 提出的 Chase 算法通过翻转最不可靠的码字后多次运行 BM 算法来实现译码。这两种算法的原理是很相似的,前者性能低,后者复杂度大。
2. **代数软判决译码算法⁴** 此类算法的代表就是 Koetter 和 Vardy 提出的 KV 算法^[2]。KV 算法通过对不同的插值点分配不同的重数,将信道输出的软信息转换成代数约束条件,之后再执行 GS⁵算法^[2],实现了软判决译码。KV 算法可以方便的调节参数,实现复杂度和性能的折中。

3. **自适应的信度传播算法⁶** ABP 算法由 Jiang 等人于 2004 年提出^[2],它利用

³GMD,Chase 算法

⁴ASD:Algebraic Soft Decision 算法

⁵由 Guruswami 和 Sudan 提出的代数列表译码算法

⁶ABP:Adaptive Belief Propagation 算法

了在 LDPC 码译码中常用的 BP 算法。BP 算法要求校验矩阵式稀疏的,而 RS 码进行二进制展开后的校验矩阵仍然是高密度的。Jiang 通过在 BP 算法前对矩阵进行一定的处理,使其对应于置信度较低的矩阵式稀疏的,而后再进行 BP 迭代,实现译码。

介绍以上方式,只是说明 RS 码应用很广泛,译码方法很成熟,但是针对于水声通信来说,不需要这么复杂的软判决译码算法,而 BM 算法的简单,易实现特点很适合于水声通信。下面就 Berlekmap-Massey 详细讲解。

4.5 Berlekmap-Massey 硬判决算法

4.5.1 Berlekamp-Massey 迭代原理^[2]

众所周知分组码最典型的硬判决译码方法就是伴随式 (Syndrome) 译码,译码分为三个步骤:

1. 计算伴随式;
2. 由伴随式求取错误图样
3. 将错误图样与接收码字相加,得到译码输出

在第二步中,又分为求取错误位置和求取错误值,而 BM 算法本质上是求取错误位置多项式,而利用钱搜索计算错误位置多项式的根,而 Forney 算法来计算错误值,这两个算法会在第四小节介绍。下面对 BM 算法的原理做些介绍^[2]。

把错误图样 \vec{e} 也看做一个多项式:

$$e(x) = e_0 + e_1x + e_2x^2 + \cdots + e_{n-1}x^{n-1} \quad (4-12)$$

假如实际发生的错误为 $v, 0 \leq v \leq t, t$ 为 RS 码纠错能力 ($n - k = 2t$),那么,错误多项式就可以写成仅有 v 项的形式:

$$e(x) = e_{i_1}x^{i_1} + e_{i_2}x^{i_2} + \cdots + e_{i_v}x^{i_v} \quad (4-13)$$

其对应的错误图样的形式为: $\vec{e} = (0 \cdots e_{i_1} \cdots 0 \cdots e_{i_2} \cdots \cdots 0 \cdots e_{i_v} \cdots 0)$, 下标 i_l 在 $0 \sim n - 1$ 整数中取值,代表错误的位置, l 在 $1 \sim v$ 取值。

知道, $\vec{V} = \vec{r}H^T = \vec{e}H^T$, 代入 RS 码 H 矩阵的形式, 则有:

$$\vec{S} = (0 \cdots e_{i_1} \cdots 0 \cdots e_{i_2} \cdots 0 \cdots e_{i_v}) \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha^1 & \alpha^2 & \cdots & \alpha^{2t} \\ (\alpha^1)^2 & (\alpha^2)^2 & \cdots & (\alpha^{2t})^2 \\ \vdots & \vdots & \vdots & \vdots \\ (\alpha^1)^{n-1} & (\alpha^2)^{n-1} & \cdots & (\alpha^{2t})^{n-1} \end{bmatrix} \quad (4-14)$$

由矩阵乘法定义可知, \vec{e}^T 和 H 的 $2t$ 个列向量进行求内积运算, \vec{e}^T 与第 j 个列向量求内积求出 \vec{S} 的第 j 个分量:

$$S_j = e_{i_1} (\alpha^j)^{i_1} + e_{i_2} (\alpha^j)^{i_2} + \cdots + e_{i_v} (\alpha^j)^{i_v} = e (\alpha^j), 1 \leq j \leq 2t \quad (4-15)$$

为了表达方便, 假设 $Y_l = e_{i_l}, 1 \leq l \leq v$; 设 $X_l = \alpha^{i_l}, 1 \leq l \leq v$, 事实上, Y_l 就是错误图样中某个错误位置上的错误值, 而 X_l 就代表着错误位置 i_l , 这样, 上式 4-15 可以重写为:

$$S_j = Y_1 X_1^j + Y_2 X_2^j + \cdots + Y_v X_v^j, 1 \leq j \leq 2t \quad (4-16)$$

这可以看作一个方程组。设接收码字为 $r(x) = c(x) + e(x)$, 因为 $\alpha^j, 1 \leq j \leq 2t$ 是 $c(x)$ 的根: $c(\alpha^j) = 0$, 由式 4-15, $r(\alpha^j) = e(\alpha^j) = S_j$, 接收码字在接收端已知, 可以求出 $2t$ 个伴随式 S_j , 即完成伴随式译码的第一步。式 4-16 中未知的是 X_l, Y_l 的值, 共 $2v$ 个。 $v \leq t$, 可以通过解该方程组求得错误位置和错误值, 但是这是个非线性方程组, 求解过程非常复杂, 因此采用先求错误位置, 再求错误值。

定义次数为 v 错误位置多项式 $\sigma(x)$:

$$\sigma(x) = 1 + \sigma_1 x + \sigma_2 x^2 + \cdots + \sigma_v x^v = (1 - X_1 x)(1 - X_2 x) \cdots (1 - X_v x) \quad (4-17)$$

这个定义表明, 错误位置 X_l 的逆元 X_l^{-1} 是 $\sigma(x)$ 的根。下面, 目标就变成求 $\sigma(x)$ 及其根。

可以证明^[2], $\sigma(x)$ 的系数与伴随式 S_j 之间存在如下关系:

$$\begin{bmatrix} S_1 & S_2 & \cdots & S_v \\ S_2 & S_3 & \cdots & S_{v+1} \\ \vdots & \vdots & \vdots & \vdots \\ S_v & S_{v+1} & \cdots & S_{2v-1} \end{bmatrix} \begin{bmatrix} \sigma_v \\ \sigma_{v-1} \\ \vdots \\ \sigma_1 \end{bmatrix} = - \begin{bmatrix} S_{v+1} \\ S_{v+2} \\ \vdots \\ S_{2v} \end{bmatrix} \quad (4-18)$$

而 Berlekamp 提出了一种基于上式4-18迭代求 $\sigma(x)$ 的方法, Massey^[2] 发现了这种迭代算法和线性反馈移位寄存器的关系, 因此, 把这种实用的求解算法称为 BM 算法, 下面研究 BM 迭代^[2] 求 $\sigma(x)$ 的具体过程, 式4-18的关系可以简写成:

$$S_j = - \sum_{i=1}^v \sigma(x)_i S_{j-i} \quad j = v+1, \dots, 2v \quad (4-19)$$

第二章讲过, 在 $GF(2^m)$ 上的加法求逆和加法等价, 因此, 可以把负号可以省略。

式4-19可以等效成一个线性反馈移位寄存器 (设 $v = 3$):

图 4-3 BM 迭代过程移位寄存器图

如果 $\sigma_1, \dots, \sigma_v$ 已知, 将寄存器中置入 S_1, \dots, S_v 的值, 随着移位, 在输出处就会产生 S_{v+1}, \dots, S_{2v} 的值。现在的问题是, S_1, \dots, S_{2t} 是知道的, $\sigma_1, \dots, \sigma_v$ 未知。可以采用迭代的方法构造出一个移位寄存器来符合上面的输入输出关系。注意到, $\sigma(x)$ 的系数就是寄存器的抽头权重。

可以用数学归纳法完成对迭代过程的推导。假设, 第 $r-1$ 次迭代, 寄存器长度 L_{r-1} , 此时的寄存器抽头权重为 $\sigma^{r-1}(x)$, 它能够保证产生准确的 S_1, \dots, S_{r-1} 。下面的问题是怎么产生 $\sigma^r(x)$ 能够使它能产生 S_1, \dots, S_r 。当抽头权重为 $\sigma^{r-1}(x)$, 前 $r-1$ 次移位, 产生了 S_1, \dots, S_{r-1} , 第 r 次移位, 输出端的值记作 S'_r :

$$S'_r = - \sum_{j=1}^{D(r-1)} \sigma_j^{r-1} S_{r-j} \quad (4-20)$$

其中 $D(r-1) = \deg \sigma^{r-1}(x)$ 。它可能不等于 S_r , 定义它们的差值:

$$\Delta_r = S_r - S'_r = S_r + \sum_{j=1}^{D(r-1)} \sigma_j^{(r-1)} S_{r-j} = \sum_{j=0}^{D(r-1)} \sigma_j^{(r-1)} S_{r-j} \quad (4-21)$$

如果 $\Delta_r = 0$, 说明 $\sigma^{r-1}(x)$ 也能够正确地产生 S_r , 那么 $\sigma^r(x) = \sigma^{r-1}(x)$, 否则, 要对抽头系数做个修正:

$$\sigma^{(r)} = \sigma^{(r-1)}(x) + Ax^l \sigma^{(m-1)}(x) \quad (4-22)$$

这里的 $\sigma^{(m-1)}(x)$ 是前面 $r-1$ 次迭代里出现过的一个多项式, l 是一个整数, A 是一个有限域元素, 它们的选择我们下面再分析, 选择它们的依据就在于它们的值能够让系数修正以后的 $\Delta'_r = 0$ 。

对这个新的多项式 $\sigma^r(x)$, 它的输出和 S_r 的差值为:

$$\Delta'_r = \Delta_r + A \sum_{j=0}^{D(m-1)} S_{r-j-l} \quad (4-23)$$

可以选择前面迭代过程中出现过的一个多项式 $\sigma^{m-1}(x)$, 满足 $\Delta_m \neq 0$, 令 $l = r - m$, 则上式4-23第二项变成了 $A\Delta_m$, 再选择 $A = -\Delta_r/\Delta_m$, 则有:

$$\Delta'_r = \Delta_r - \frac{\Delta_r}{\Delta_m} \Delta_m = 0 \quad (4-24)$$

这样就满足了 $\Delta'_r = 0$ 的条件, 即: $\sigma^r(x)$ 能够产生 S_r 。可以证明, S_1, \dots, S_r 同也能够由 $\sigma^r(x)$ 产生, 事实上, 式4-22可以看作把 $\sigma^{m-1}(x)$ 连接在 $\sigma^{r-1}(x)$ 上的辅助寄存器, 当 $\sigma^r(x)$ 进行前 $r-1$ 移位时, 它始终产生 0, 只有在第 r 次移位时, 产生一个非零值 Δ_m , 经过系统 A 修正后, 正好用来补偿 $\sigma^{r-1}(x)$ 的输出, 使其输出 S_r , 这就是通过修正的方法产生 $\sigma^r(x)$ 的基本思想。

如果只按照上面的说明, 这样的 $\sigma^{m-1}(x)$ 不是唯一的, 这是不允许的。在 BM 算法中有一个约束条件^[2], 就是 $\sigma^r(x)$ 的次数要尽可能低, 因为 $\sigma(x)$ 的次数就是错误的个数, 在译码时总认为错误少的概率比错误多的概率高。事实上, 在陪集首译码中作为陪集首的错误图样也要求重量尽可能轻, 也是一样的道理。这些都是为了保证我们的译码是最小距离译码, 即选择于接收码字汉明距离最小的码字作为译码输出。为了满足这个条件, 在选择 m 的时候, 需要选择的是出现在 $\sigma^{r-1}(x)$ 前面, 刚刚改变长度的那个 $\sigma^{m-1}(x)$, 即 $L_{r-1} = L_M > L_{m-1}$ 。这样可以保证产生的 $\sigma^r(x)$ 始终次数最低。经过这样的迭代, 迭代 $2t$ 次, 就可以得到最终的错误位置多项式 $\sigma^r(x)$ 。

4.5.2 BM 算法的主要步骤及实现^[2]

下面,给出 BM 迭代的流程图4-4。流程图中的 $B(x)$ 相当于前面说的 $\sigma^{m-1}(x)$, 两者只相差一个系数,而这个系数实际上是原来系数 A 里面分配了义部分。计算 $T(x)$ 的多项式实际上就是式4-20。具体步骤分析如下:

图 4-4 BM 迭代流程图

1. 初始化: $\sigma(x)$, $r = 0$, $L = 0$, $B(x) = 1$, 然后开始迭代 $r = r + 1$;
2. 按 $\Delta_r = S_r + \sum_{j=1}^{D(r-1)} \sigma_j^{(r-1)} S_{r-j}$ 计算 Δ_r , 若 $\Delta_r = 0$, 则有 $B(x) = xB(x)$, 并计算 Δ_{r+1} , 再进行下一次迭代;

3. 若 $\Delta_r \neq 0$, 计算修正后的多项式 $T(x) = \sigma(x) - \Delta_r x B(x)$, 若 $2L \leq r - 1$ 不成立, 则 $\sigma(x) = T(x)$, $B(x) = xB(x)$, 如果成立, 继续;
4. 保存 $B(x)$ 用于下一次修正 $B(x) = \Delta_r^{-1} \sigma(x)$, $\sigma(x) = T(x)$, $L = r - L$, 判断 $r = 2t$, 如果否, 再进行下一次迭代; 如果是, 继续;
5. 如果 $\sigma(x)$ 的最高次数小于等于 t 那么纠错完成, 否则, 不可纠错。

4.5.3 钱搜索与 Forney 算法

求得 $\sigma(x)$, 下一步的问题就是求得多项式的根。工程上使用钱搜索方法来确定错误位置, 钱搜索的原理在于有限域的元素是有限域的。可以逐个代入进行验证。这部分比较容易理解, 在此做简要说明

$$\begin{aligned}\sigma(x) &= (1 - X_1 x)(1 - X_2 x) \cdots (1 - X_v x) \\ &= \prod_{l=1}^v (1 - X_l x) = \sigma_v x^v + \sigma_{v-1} x^{v-1} + \cdots + \sigma_1 x + \sigma_0\end{aligned}\quad (4-25)$$

首先验证位置 x^0 是否有错误, 即把 $x = \frac{1}{\alpha^0} = \alpha^n (\alpha^n = 1)$ 代入 $\sigma(x)$ 若结果等于 0, 说明第一个码元有错误, 否则第一个码元是正确的。

把 $x = \frac{1}{\alpha^1} = \alpha^{n-1}$ 代入 $\sigma(x)$, 验证位置 x^1 是否有错误。同理把 $\alpha^{n-2}, \alpha^{n-3}, \dots, \alpha$ 代入 $\sigma(x)$ 分别验证位置 x^2, \dots, x^{n-1} 是否错误。

下面还要求取错误值 Y_l , 这里利用的是 Forney 算法。定义错误值多项式:

$$\Omega(x) = S(x)\sigma(x) \mod x^{2t} \quad (4-26)$$

这个式子又可以称作关键方程, 其中 $S(x) = 1 + S_1 x + S_2 x^2 + \cdots + S_{2t} x^{2t}$, 通过上面的步骤, 求得了 $S(x)$ 和 $\sigma(x)$, 因此 $\Omega(x)$ 可以很快求出, 而 $\Omega(x)$ 还可以写为^[2]:

$$\Omega(x) = x \sum_{i=1}^v Y_i X_i \prod_{l \neq i} (1 - X_l x) \quad (4-27)$$

将 $\sigma(x)$ 的某个根 X_l^{-1} 代入, 就可以求出对应位置的错误值:

$$Y_l = \frac{\Omega(X_l^{-1})}{\prod_{j \neq l} (1 - X_j X_l^{-1})} \quad (4-28)$$

这样, 错误图样就被确定, 将接收码字和错误图样相加, 就得到了译码码字。

4.5.4 BM 算法性能仿真

通过 matlab 仿真信道特性以及调制方式,而 C 语言实现编译码方式,最后得到的数据在 matlab 里编程画出误码率曲线图。

图4-5是 $RS(15, 9, 7)$ 码在 PSK 调制方式下的误码率仿真图。

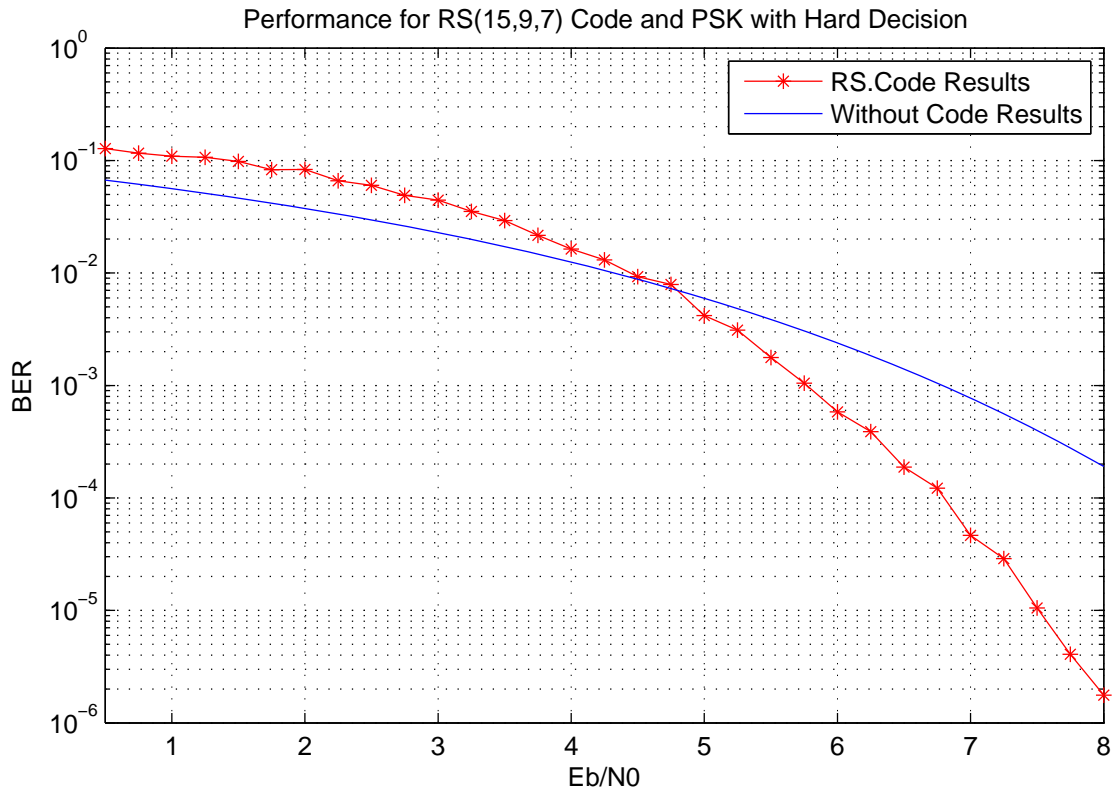


图 4-5 $RS(15, 9, 7)$ 在 PSK 调制下误码率曲线图

从图4-5可以看出:RS 码 BM 译码算法比未编码的信道误码率要好的多。

E_b/N_0 低于 4.75 的时候,未编码比编码效果好,这是一种误解,因为 RS 码是多进制码,而本设计采用的是四进制,因此,在相同信号幅值和噪声的情况下,未编码码的 E_b/N_0 是 RS 码的四倍,即未编码在 $E_b/N_0 = 1$ 的误码率应该还 RS 码 $E_b/N_0 = 4$ 的误码率相比,从图中可以看出,还是 RS 码误码率低。

4.6 本章小结

本章从 RS 码概述,编码器,译码器设计等方面介绍了 RS 码,其中编码器讨论了乘法形式的编码器和除法形式的编码器,而我们采用的是基于 $h(x)$ 除法形式的编码器,因为,这种方式编出来的码字是系统码,然后从理论和具体实现步骤上介绍了 BM 算法,当然,为了译码完整,还简单介绍了用于求错误位置的钱搜索和求错误值的 Forney 算法,最后仿真验证性能。

第五章 EXIT 图在 Turbo 均衡中的应用

5.1 PC/104 简介

PC/104(pc104) 是 ISA(IEEE-996) 标准的延伸。1992 年 PC/104 作为基本文件被采纳,叫做 IEEE-P996.1 兼容 PC 嵌入式模块标准。PC/104 是一种专门为嵌入式控制而定义的工业控制总线。IEEE-P996 是 ISA 工业总线规范,IEEE 协会将它定义 IEEE-P996.1,PC/104 实质上就是一种紧凑型的 IEEE-P996,其信号定义和 PC/AT 基本一致,但电气和机械规范却完全不同,是一种优化的、小型、堆栈式结构的嵌入式控制系统。其小型化的尺寸($90 \times 96\text{mm}$),极低的功耗(典型模块为 1 — 2 瓦)和堆栈的总线形式(决定了其高可靠性),受到了众多从事嵌入式产品生产厂商的欢迎,在嵌入式系统中领域逐渐流行开来。

PC/104 的优点

- 大尺寸

PC/104 的板卡标准尺寸为 $90 \times 96\text{mm}$ (比一本新华字典还要小很多,而传统桌面 PC 系统的板卡尺寸为 $315 \times 122\text{mm}$),这样小的尺寸使得 PC/104、PC/104+ 和 PCI-104 模块板成为了嵌入式系统应用的理想产品。

- 开放的高可靠性的工业规范

PC/104、PC/104+ 和 PCI-104 产品在电气特性和机械特性上可靠性极高,功耗低,产生热量少。板卡与板卡之间通过自堆栈进行可靠的连接,抗震能力强。全世界有超过 200 家公司使用这些开放的规范来生产和销售各种 PC/104 模块板。

- 模块可自用扩展

PC/104 模块具有灵活的可扩展性。它允许工程师互换及匹配各种功能卡,可随系统的需求而升级 CPU 的性能。增加系统的功能和性能只需通过改变相应的模块即可实现。

- 低功耗

4mA 的总线驱动电流,即可使模块正常工作,低功耗有利于减少元件数量。

各种插卡广泛采用 VLSI 芯片、低功耗的 ASIC 芯片、门阵列等,其存储采用大容量固态盘(SSD)。

• 堆栈式连接

这种结构取消了主板和插槽,可以将所有的 PC/104 模块板利用板上的叠装总线插座连接起来。有效减小整个系统所占的空间。PC104 板的叠装总线插座是针脚插接方式,理论上可以无限扩插 N 多扩展卡,但要看他的承受能力。

5.2 卷积码序贯译码在 PC/104 上运行

在 PC/104 上,测试 $POLY1 = 0xA5048D$, $POLY2 = 0xDAFB73$ 约束长度为 $K = 24$ 的卷积码的运行效率与误码率,下表就是测试的一些数据,其中测试的数据长度为 $L = 1000000$ 比特,既是 125000 字节,又根据第三章编码中,终结码介绍可知,加上 24 比特,这样可以实现结尾的误码率不受结尾码字的影响。测试环境为 BPSK 调制,AWGN 信道。表5-1就是运行得到的数据。从表5-1,结合参考书

表 5-1 约束长度 $K = 24$ 的卷积码测试数据表

E_b/N_0	2.0	2.25	2.5	2.75	3.0	3.25
比特错误率	1.001×10^{-1}	8.93×10^{-2}	6.24×10^{-2}	2.65×10^{-2}	9.5×10^{-3}	5.8×10^{-3}
E_b/N_0	3.50	3.75	4.0	4.25	4.5	4.75
比特错误率	1.7×10^{-3}	7.8×10^{-4}	3.12×10^{-4}	1.15×10^{-4}	1.0×10^{-5}	1.01×10^{-6}

上给出的实例可以看出,误码率还是符合理论要求的。因此,现在要测试的就是运行的时间,译码运行时间的大小,决定译码时候缓冲区有无以及大小。表5-2将给出在各个 E_b/N_0 下,译码器运行的时间,其中码长为 1000024 比特。从图中可以看出,序贯译码,在 E_b/N_0 很低的时候,运行时间需要很长,这是因为,低 E_b/N_0 情况

表 5-2 约束长度 $K = 24$ 的卷积码运行时间表

E_b/N_0	2.0	2.25	2.5	2.75	3.0	3.25
运行时间 (s)	59.16	50.87	46.39	44.49	40.34	39.50
E_b/N_0	3.50	3.75	4.0	4.25	4.5	4.75
运行时间 (s)	39.99	35.45	33.57	32.90	30.00	28.5

下,序贯译码需要大量的运算,因此回溯次数会增加。而 E_b/N_0 越高相对需要的时间越少。

从总体考虑,缓冲区还是需要的,尤其是传输码率和数据量很大的时候,至于缓冲区大小的确定这要根据数据量的多少,码字传输速率, E_b/N_0 等情况综合考虑。

5.3 RS 码 BM 算法译码在 PC/104 上运行

在 PC/104 上,测试 $RS(15, 9, 7)$ 的运行效率与误码率,下表就是测试的一些数据,错误的帧数为 200,且每帧 36 比特,高于此帧数的时候就停止输入数据,停止编译码。测试环境为 BPSK 调制,AWGN 信道。

表 5-3 $RS(15, 9, 7)$ 测试数据表

E_b/N_0	0.5	0.75	1	1.25	1.5	1.75
比特错误率	1.3×10^{-1}	1.2×10^{-1}	1.0×10^{-1}	1.0×10^{-1}	9.8×10^{-2}	8.3×10^{-2}
字节错误率	6.4×10^{-1}	6.2×10^{-1}	6.0×10^{-1}	5.8×10^{-1}	5.2×10^{-1}	4.7×10^{-1}
帧错误率	9.4×10^{-1}	9.0×10^{-1}	8.9×10^{-1}	8.7×10^{-1}	8.5×10^{-1}	7.6×10^{-1}

E_b/N_0	2	2.25	2.5	2.75	3	3.25
比特错误率	8.3×10^{-2}	6.0×10^{-2}	6.1×10^{-2}	4.9×10^{-2}	4.4×10^{-2}	3.5×10^{-2}
字节错误率	4.6×10^{-1}	3.7×10^{-1}	3.5×10^{-1}	3.0×10^{-1}	2.7×10^{-1}	2.2×10^{-1}
帧错误率	7.5×10^{-1}	6.4×10^{-1}	5.9×10^{-1}	5.2×10^{-1}	4.7×10^{-1}	3.9×10^{-1}

续下页

续表 5-3 $RS(15, 9, 7)$ 测试数据表

E_b/N_0	3.5	3.75	4	4.25	4.5	4.75
比特错误率	2.9×10^{-2}	2.2×10^{-2}	1.6×10^{-2}	1.3×10^{-2}	9.3×10^{-3}	7.9×10^{-3}
字节错误率	1.8×10^{-1}	1.3×10^{-1}	1.1×10^{-1}	8.6×10^{-2}	6.0×10^{-2}	5.2×10^{-2}
帧错误率	3.2×10^{-1}	2.4×10^{-1}	2.0×10^{-1}	1.6×10^{-1}	1.1×10^{-1}	9.9×10^{-2}

E_b/N_0	5	5.25	5.5	5.75	6	6.25
比特错误率	4.2×10^{-3}	3.1×10^{-3}	1.8×10^{-3}	1.1×10^{-3}	5.8×10^{-4}	3.9×10^{-4}
字节错误率	2.9×10^{-2}	1.9×10^{-2}	1.2×10^{-2}	7.3×10^{-3}	3.9×10^{-3}	2.6×10^{-3}
帧错误率	5.5×10^{-2}	3.7×10^{-2}	2.3×10^{-2}	1.4×10^{-2}	7.5×10^{-3}	5.0×10^{-3}

E_b/N_0	6.5	6.75	7	7.25	7.5	7.75
比特错误率	1.9×10^{-4}	1.2×10^{-4}	4.6×10^{-5}	2.9×10^{-5}	1.1×10^{-5}	4.1×10^{-6}
字节错误率	1.3×10^{-3}	8.1×10^{-4}	3.1×10^{-4}	1.9×10^{-4}	6.8×10^{-5}	2.8×10^{-5}
帧错误率	2.4×10^{-3}	1.6×10^{-3}	6.1×10^{-4}	3.6×10^{-4}	1.4×10^{-4}	5.7×10^{-5}

表5-3测试了 E_b/N_0 从 2 到 7.75 步进为 0.25 的比特错误率,字节错误率和帧错误率,表格很细致的反映了 RS 误码率的走势和未经过信道编码的 BPSK 调制误码率的比较,从而可以看出,RS 还是很好的降低了误码率,满足了我们的要求。

表 5-4 $RS(15, 9, 7)$ 运行时间表

E_b/N_0	0.5	0.75	1	1.25	1.5	1.75
运行时间 (s)	1.0×10^{-2}	1.0×10^{-2}	1.0×10^{-2}	1.0×10^{-2}	1.0×10^{-2}	2.0×10^{-2}
总比特数	7.7×10^3	8.0×10^3	8.1×10^3	8.4×10^3	8.5×10^3	9.6×10^3

E_b/N_0	2	2.25	2.5	2.75	3	3.25
运行时间 (s)	1.0×10^{-2}	2.0×10^{-2}	2.0×10^{-2}	1.0×10^{-2}	3.0×10^{-2}	2.0×10^{-2}

续下页

续表 5-4 RS(15, 9, 7) 运行时间表

总比特数	9.7×10^3	1.1×10^4	1.2×10^4	1.4×10^4	1.5×10^4	1.9×10^4
------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------

E_b/N_0	3.5	3.75	4	4.25	4.5	4.75
运行时间 (s)	4.0×10^{-2}	3.0×10^{-2}	7.0×10^{-2}	5.0×10^{-2}	9.0×10^{-2}	1.2×10^{-1}
总比特数	2.2×10^4	3.0×10^4	3.6×10^4	4.4×10^4	6.5×10^4	7.3×10^4

E_b/N_0	5	5.25	5.5	5.75	6	6.25
运行时间 (s)	1.7×10^{-1}	2.8×10^{-1}	4.2×10^{-1}	6.7×10^{-1}	1.2×10^0	2.1×10^0
总比特数	1.3×10^5	1.9×10^5	3.1×10^5	5.1×10^5	9.6×10^5	1.4×10^6

E_b/N_0	6.5	6.75	7	7.25	7.5	7.75
运行时间 (s)	3.4×10^0	7.4×10^0	1.4×10^1	3.4×10^1	7.1×10^1	1.7×10^2
总比特数	2.8×10^6	4.5×10^6	1.2×10^7	2.0×10^7	5.3×10^7	1.3×10^8

从表5-4中可以看出,随着 E_b/N_0 的升高,运行时间一直在增加,这里有个误解,事实上并非如此,由于随着 E_b/N_0 的升高,误码率是下降的,为了精确度的原因,必然需要越多的数据进行测试,才能达到误码率精度的要求,因此随着 E_b/N_0 的升高,测试数据也增多,相应的运行时间也是增多的。

通过表中运行时间和测试数据量的关系可以看出,运行时间和 E_b/N_0 并无关系,这也是理论上支持的。而且从运行时间上看,译码算法还是很高效的,在加上本设计的目的和水声通信的要求,RS 编译码只用于少量数据的传输,因此缓冲区可以省掉。

5.4 本章小结

本小节首先介绍了 PC/104,并说明为什么选择 PC/104 作为我们测试译码算法的平台,其次在给出了序贯译码算法和 BM 译码算法在 PC/104 上运行的结果,包括测试误码率数据和运行时间,以及讨论关于缓冲区的选取和大小的选择。

第六章 水声通信系统 Turbo 均衡算法测试数据分析

6.1 研究工作总结

由于人们对海洋的关注越来越多,人类在海洋中的活动越来越频繁,因此作为海洋技术基础的水声通信,必然也是人们关注的焦点,而信道编码作为水声通信系统可靠性的保障,也必然研究广泛。

本文分别从编码理论、卷积码编译码以及 RS 码编译码三个方面介绍水声通信的信道编码。

1. 编码代数理论,作为信道编码的基础,有必要认真的学习一番,这是我们理解和应用各种信道编码的前提,因此在最初,就介绍了信道编码的相关代数基础。
2. 卷积码的编译码,从原理和实现两个方面介绍了整个编译码过程,着重讨论了序贯译码方式的费诺算法,并仿真和分析性能。
3. RS 码的编译码,也是从原理和实现两个方面介绍了整个编译码过程,着重讨论了 BM 译码算法,并仿真和分析性能。

6.2 下一步工作

6.2.1 费诺算法的进一步改进

在上述章节中提到的费诺算法,我们知道,其运算量很大,尤其是在信噪比很低的时候,因此会有很大的延时,这是系统不能容忍的,而与之对应的另外一种序贯译码算法——多堆栈算法,却能避免大量的运算,只是存储空间的浪费比费诺算法严重的多。

结合这两种算法,使其具有多堆栈算法的效率,又有费诺算法的节省存储空间。

6.2.2 Turbo 码等其他码字的研究

由于水声通信和编码技术的日益发展,其他一些新的码字也渐渐地显示出自己的优势,而 Turbo 无疑是其中的佼佼者,因此,Turbo 码也是下一步工作的重点。

最后还有级联码,因为已经做好了卷积码和 RS 码的编译码,因此把 RS 码作为外码,卷积码作为内码的级联码的仿真也是一个重点。

参考文献

- [1] 栾维新. 中国海洋产业高技术化研究. 海洋出版社, 北京.
- [2] G.Proakis. *Digital Communication*. McGraw-Hill, New York.
- [3] 林宇航. 中微子通信:通信技术史上的一场新技术革命. *数字通信*, 13(1):8--9, January 1996.
- [4] 路轶群. 蓝绿激光通信及探潜. *光电子技术与信息*, 13(2):1--7, February 1991.
- [5] 李启虎. 水声学研究进展. *声学学报*, 26(4):295--301, 2001.
- [6] 李启虎. 进入 21 世纪的声纳技术. *应用声学*, 21(1):13--18, January 2002.
- [7] WA Kuperman, William S Hodgkiss, Hee Chun Song, T Akal, C Ferla, and Darrell R Jackson. Phase conjugation in the ocean: Experimental demonstration of an acoustic time-reversal mirror. *The journal of the Acoustical Society of America*, 103:25, 1998.
- [8] M. Stojanovic and L. Freitag. Wideband underwater acoustic cdma: adaptive multichannel receiver design. In *OCEANS, 2005. Proceedings of MTS/IEEE*, pages 1508--1513, sept. 2005.
- [9] T Xu and K. Xu. Research on high data rate transmission in shallow water acoustic channel. In *International Conference on Shallow-Water Acoustics*, April. 1997.
- [10] M. Stojanovic. Recent advances in high-speed underwater acoustic communications. *Oceanic Engineering, IEEE Journal of*, 21(2):125--136, apr 1996.
- [11] J.A. Catipovic. Performance limitations in underwater acoustic telemetry. *Oceanic Engineering, IEEE Journal of*, 15(3):205--216, jul 1990.
- [12] B. Woodward and H. Sari. Digital underwater acoustic voice communications. *Oceanic Engineering, IEEE Journal of*, 21(2):181--192, apr 1996.
- [13] M. Stojanovic and Z. Zvonar. Multichannel processing of broad-band multiuser communication signals in shallow water acoustic channels. *Oceanic Engineering, IEEE Journal of*, 21(2):156--166, apr 1996.
- [14] M. Stojanovic, J.A. Catipovic, and J.G. Proakis. Phase-coherent digital communi-

- cations for underwater acoustic channels. *Oceanic Engineering, IEEE Journal of*, 19(1):100--111, jan 1994.
- [15] E.M. Sozer, M. Stojanovic, and J.G. Proakis. Underwater acoustic networks. *Oceanic Engineering, IEEE Journal of*, 25(1):72--83, jan. 2000.
- [16] J. Rice, B. Creber, C. Fletcher, P. Baxley, K. Rogers, K. McDonald, D. Rees, M. Wolf, S. Merriam, R. Mehio, J. Proakis, K. Scussel, D. Porta, J. Baker, J. Hardiman, and D. Green. Evolution of seaweb underwater acoustic networking. In *OCEANS 2000 MTS/IEEE Conference and Exhibition*, volume 3, pages 2007--2017, 2000.
- [17] Zhu Wei-Qing, Wang Chang-hong, Pan Feng, Zhu Min, Wang Rui, Zhang Xiang-Jun, and Dai Yong-Mei. Underwater acoustic communication system of auv. In *OCEANS '98 Conference Proceedings*, volume 1, pages 477--481, sep-1 oct 1998.
- [18] Y. Gu and T. Le-Ngoc. Adaptive combined DFE/MLSE techniques for ISI channels. *IEEE Trans. Commun.*, 44(7):846--847, 1996.
- [19] Y. Liu and S.D. Blostein. Identification of frequency non-selective channels using decision feedback and adaptive linear prediction. *IEEE Trans. Commun.*, 42(2):1484--1492, 1995.
- [20] X.Y. Yu and S. Pasupathy. Innovations-based MLSE for rayleigh fading channels. *IEEE Trans. Commun.*, 43(2):1534--1544, 1995.
- [21] B. Hart and D.P. Taylor. Extended MLSE Diversity Receiver for the Time and Frequency Selective Channel. In *Communications, 1996. ICC '96, Conference Record, Converging Technologies for Tomorrow's Applications. 1996 IEEE International Conference on*, 1996.
- [22] L. Yin and G.L. Stuber. MLSE and Soft-Output Equalization for Trellis-Coded Continuous Phase Modulation. *IEEE Trans. Commun.*, 45(6):651--658, 1997.
- [23] W. Younis and N. Al-Dhahir. Joint prefiltering and MLSE equalization of space-time coded transmissions over frequency-selective channels. *IEEE Trans. Veh. Technol.*, pages 144--154, 2002.

- [24] T. Wang and C.W. ng. On Adaptive Decission-Feedback Equalization of Inter symbol Interference Channels in Coded Modulation Systems. *IEEE Trans. Commun.*, 44(11):1404--1412, 1996.
- [25] Proakis J.G. Zhou, K. and L. Ling. Decision-Feedback Equalization of Time-Dispersive Channels with Coded Modulation. *IEEE Trans. Commun.*, 38(1):18--24, 1990.
- [26] T.C. Yang. Differences Between Passive-Phase Conjugation and Decision-Feedback Equalizer for Underwater Acoustic Communications. *IEEE Journal of Oceanic Engineering*, 29(2):472--487, April 2004.
- [27] P Magniez and Cergy-pontoise Cedex. Turbo-Equalization applied to Trellis-Coded-Modulations. pages 2556--2560, 1999.
- [28] Dan Raphaeli and Zarai Yoram. Combined Turbo Equalization And Turbo Decoding. In *Global Telecommunications Conference, 1997. GLOBECOM '97., IEEE*, volume 2, pages 0--4, 1997.
- [29] S Vlahoyiannatos, S X Ng, and L Hanzo. Combineed Trellis-Coded Modulation and Blind Turbo Equalisation. In *EUROCON'2001, Trends in Communications, International Conference on.*, pages 508--512, 2001.
- [30] Equalization Combined, With Trellis, Turbo Trellis, Coded Modulation, and Satellite Channel. EQUALIZATION COMBINED WITH TRELLIS CODED AND TURBO TRELLIS CODED MODULATION IN THE NONLINEAR Sk FtJT. 00(C):0--4, 2000.
- [31] Michael Tüchler, Ralf Koetter, and Andrew C Singer. Turbo Equalization : Principles and New Results. *Communications, IEEE Transactions on*, 50(5):754--767, 2002.
- [32] Michael Tüchler and Andrew C Singer. Turbo Equalization : An Overview. *IEEE Transactions on Information Theory*, 57(2):920--952, 2011.
- [33] Michael Tüchler, Andrew C Singer, and Ralf Koetter. Minimum Mean Squared Error Equalization Using A Priori Information. *IEEE Transactions on Signal Pro-*

- cessing, 50(3):673--683, 2002.
- [34] Cheng Xiang, Yuan Dongfeng S, and Y U Xiangming. The Improvement of Turbo Equalization through Using Turbo Codes. In *Communications, Circuits and Systems, 2005. Proceedings. 2005 International Conference on*, number 60372030, pages 124--127, 2003.
- [35] P Yang and J Ge. Combination of turbo equalisation and Turbo Trellis-coded modulation with low complexity. (4):772--775, 2007.
- [36] Peng Yang and Jian-hua Ge. Combination of turbo equalization and turbo TCM for mobile communication system. *IEEE International Symposium on Communications and Information Technology, 2005. ISCIT 2005.*, 1(1):370--373, 2005.
- [37] A Anastasopoulos and KM Chugg. Iterative equalization/decoding of TCM for frequency-selective fading channels. In *Signals, Systems & Computers, 1997. Conference Record of the Thirty-First Asilomar Conference on*, pages 177--181, 1997.
- [38] L. Hanzo, T.H. Liew, and B.L. Yeap. *Turbo Coding, Turbo Equalisation and Space-Time Coding for Transmission over Fading Channel*. John Wiley & Sons, Ltd, England, 2002.
- [39] Koetter Ralf, C. Singer Andrew, and Tuchler Michael. Turbo Equalization An iterative equalization and decoding technique for coded data transmission. *IEEE SIGNAL PROCESSING MAGAZINE*, (January 2004):67--80, 2004.
- [40] C. Berrou, A. Glavieux, and P. Thitimajshima. Near Shannon limit error-correction coding and decoding: Turbo-codes. In *Communications, 1993. ICC '93 Geneva. Technical Program, Conference Record, IEEE International Conference on*, pages 1064--1070, may 1993.
- [41] Catherine Douillard, Michel Jézéquel, Claude Berror, Snnie Picart, Pierre Didier, Alain Gavieux, et al. Iterative correction of intersymbol interference: Turbo Equalization. *European Transactions on Telecommunications*, 6(5):507--511, January 1995.
- [42] Sirikiat Ariyavisitakul and Ye Li. Joint coding and decision feedback equalization

- for broadband wireless channels. In *Vehicular Technology Conference, 1998. VTC 98. 48th IEEE*, volume 3, pages 2256--2261. IEEE, 1998.
- [43] Hangenauer Joachim and Hoeher Peter. A Viterbi Algorithm with SOft-Decision Outputs and its Applications. In *IEEE Global Telecommunications Conference*, pages 1680--1686, Dallas, Texas, 1989.
- [44] M. Marandian and M. Salehi. Low complexity iterative decision feedback equalizer for 8PSK modulation in time dispersive channels. *12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications. PIMRC 2001. Proceedings (Cat. No.01TH8598)*, 1:A--102--A--106.
- [45] Xiaodong Wang and H Vincent Poor. Iterative (Turbo) soft interference cancellation and decoding for coded CDMA. *IEEE Transactions on Communications*, pages 1046--1061, 1999.
- [46] Alain Glavieux, Christophe Laot, and Joël Labat. Turbo equalization over a frequency selective channel. In *Proc. Int. Symp. Turbo Codes*, pages 96--102, 1997.
- [47] D Raphaeli and A Saguy. Linear equalizers for Turbo equalization: A new optimization criterion for determining the equalizer taps. In *Proc. 2nd Intern. Symp. on Turbo Codes, Brest, France*, pages 371--374, 2000.
- [48] Zining Eu. Turbo decision aided equalization for magnetic recording channels. *CODing and Iterative Detection for Magnetic Recording Channels*, 0:71--102, 2000.
- [49] A. Berthet, R. Visoz, and P. Tortelier. Sub-optimal turbo-detection for coded 8-PSK signals over ISI channels with application to EDGE advanced mobile system. In *Personal, Indoor and Mobile Radio Communications, 2000. PIMRC 2000. The 11th IEEE International Symposium*, volume 1, pages 151--157, 2000.
- [50] Michael Tuchler and Joachim Hagenauer. "Turbo Equalization" using frequency domain equalizers.
- [51] A. Essebbar and G. Loubet. Underwater Acoustic Channel Simulations for Communication. *IEEE Proceedings of Oceans 94*, 8(3):495--500, 1994.

- [52] Simon Haykin. *Adaptive Filter Theory(Fourth Edition)*. Prentice Hall, New Jersey.
- [53] P. Bragard and G. Jourdain. A fast self-optimized lms algorithm for non-stationary identification: application to underwater equalization. In *Acoustics, Speech, and Signal Processing, 1990. 1990 International Conference on*, volume 3, pages 1425-1428, apr. 1990.
- [54] Kostas Berberidis George-Othon Glentis and Sergios Theodoridis. Efficient Least Squares Adaptive Algorithms for Fir Transversal Filtering. In *IEEE Signal Processing Magazine*, Jul 1999.
- [55] A. Benveniste, M. Metivier, and P. Priouret. *Adaptive Algorithms and Stochastic Approximations*. Springer Verlag, Berlin, 1990.
- [56] B. Geller, V. Capellano, J.-M. Brossier, A. Essebbar, and G. Jourdain. Equalizer for video rate transmission in multipath underwater communications. *Oceanic Engineering, IEEE Journal of*, 21(2):150--155, 1996.
- [57] J Cioffi and T Kailath. Fast, recursive-least-squares transversal filters for adaptive filtering. *IEEE Tran. on ASSP*, 32:304--337, 1984.
- [58] D. Slock and T. Kailath. Numerically stable fast transversal filters for recursive least squares adaptive filtering. *IEEE Tran. on Sig. Proc.*, 39:92--114, 1991.
- [59] D. Slock and T. Kailath. Fast transversal filters with data sequence weighing. *IEEE Tran. on ASSP*, 37(3):346--359, 1989.
- [60] D. Slock, L. Chisci, H. Lev-Ari, and T. Kailath. Modular and numerically stable fast transversal filters for multichannel and multiexperiment RLS. *IEEE Tran. on Sig. Proc.*, 40(4):784--802, 1992.
- [61] A. Benallal and A. Gilloire. Improvement of the tracking capability of the numerically stable fast RLS algorithms for adaptive filtering. *IEEE on ICASSP'89*, 2:1031--1034, 1989.
- [62] V. Capellano and G. Jourdain. Comparison of adaptive algorithms for multichannel adaptive equalizers: Application to underwater acoustic communications. In *Oceans'98 Conference Proceedings*, number 2, pages 1178--1182, 1998.

- [63] B. Geller, V. Capellano, and G. Jourdain. Equalizer for real time high rate transmission in underwater communications. *IEEE on ICASSP*, (9):3179--3182, 1995.
- [64] Stojanovic, Milica, J Catipovic, and John G Proakis. Adaptive multi-channel combining and equalization for underwater acoustic communications. *The Journal of the Acoustical Society of America*, 94:1621--1632, 1993.
- [65] Catipovic, A. Josko, and Lee E Freitag. Spatial diversity processing for underwater acoustic telemetry, sep 1996. US Patent 5,559,757.
- [66] Q. Wen and J.A. Ritcey. Spatial diversity equalization for underwater acoustic communications. In *Signals, Systems and Computers, 1992. Conference Record of The Twenty-Sixth Asilomar Conference on*, volume 2, pages 1132--1136, 1992.
- [67] R. Gooch and B. Sublett. Joint spatial and temporal equalization in a decision-directed adaptive antenna system. In *Signals, Systems and Computers, 1988. Conference Record of The Twenty-Sixth Asilomar Conference on*, pages 255--259, 1988.
- [68] P. Monsen. Theoretical and measured performance of a DFE modem on a fading multipath channel. *IEEE Trans. on Commun.*, 25(10):1144--1153, 1977.
- [69] P. Balaban and J. Salz. Optimum diversity combining and equalization in digital data transmission with applications to cellular mobile radio. *IEEE Trans. on Commun.*, 40(5):895--907, 1977.
- [70] M. Stojanovic, J. Catipovic, and J. Proakis. Adaptive receivers for underwater acoustic communications: their relation to beamforming and diversity combining. In *Proc. COMCON 4, Rhodes, Greece*, 1993.
- [71] M Stojanovic, J. Catipovic, and J. Proakis. Reduced-complexity multichannel processing of underwater acoustic communication signals. *J. Acoust. Soc. Am.*, 98(2): 961--972, 1995.
- [72] M. Kocic, D. Brady, and S. Merriam. Reduced-complexity RLS estimation for shallow-water channels. In *Proceedings of the 1994 Symposium on Autonomous Underwater Vehicle Technology*, pages 165--170, 1994.

-
- [73] Z. Liang, Z. Weiqing, and Z. Min. An Adaptive Equalization Algorithm for Underwater Acoustic Coherent Communication System. *Journal of Electronics and Information Technology*, 30(3):473--484, 2008.
- [74] Z. Weiqing, W. Changhong, and P. Feng. Underwater acoustic communication system of AUV. In *in Oceans'98 Conference Proceedings Nice*, number 1, pages 477--481, 1998.
- [75] P. Strobach. Pure order recursive least-squares ladder algorithms. In *IEEE tRans. Acoust., Speech, Signal Processing*, volume 36, pages 880--897, Aug. 1986.
- [76] Andre Fonseca dos Santos, Wolfgang Rave, and Gerhard Fettweis. *A Low-Complexity Scheduling for Turbo Equalization with Turbo Decoding*. IEEE, 2007.
- [77] S. Benedetto, D. Disalar, and F. Montorsi, G. adn Pollara. Serial Concatenation of Interleaved Codes: Performance Analysis, Design, and Iterative Decoding. *IEEE Trans. Commun.*, 44(3), 1998.
- [78] Erica L. Daly, Andrew C. Singer, Jun Won Choi, and James C. Preisig. Linear turbo equalization with precoding for underwater acoustic communications. In *2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers*, pages 1319--1323. Ieee, November 2010.

个人简历及论文发表

唐怀东,男,1988 年生于江苏省徐州市。2006-2010 年就读于北京邮电大学信息与通信工程学院,获得学士学位;2010-2013 期间就读于中国科学院声学研究所海洋声学技术实验室,获得信号与信息处理硕士学位。

硕士期间的主要工作是水声通信中 Turbo 均衡的研究。研究工作基于国家 863“水声通信网络节点及组网关键技术”课题,目的是设计适用于深海通信的信道估计与均衡算法。在整个研究过程中对已有的水声通信信道估计,相位补偿以及信道均衡算法进行了对比分析,结合课题背景选取适合的算法进行深入的研究,并提出一定的改进方法,最终形成本篇论文。

硕士生期间发表的学术文章:

1. Huaidong Tang, Min Zhu, Yanbo Wu, Lijun Xu, Zeping Xing. Inter-frame Inter-leaved Bi-SOVA Algorithm for Underwater Acoustic Communication, The second International Conference on Consumer Electronics, Communications and Networks, 2012, 3:1865-1869
2. 唐怀东,朱敏,武岩波. 一种水声通信 Turbo 均衡中的软迭代信道估计算法, 电子与信息学报,

致谢

本论文的研究工作是在我的导师朱敏研究员的悉心指导下完成的。朱老师做事严谨、知识丰富且待人和善。作为科研工作者,朱老师有渊博的知识和丰富的经验;作为项目负责人,大到总体框架,小到一个小的电路板的设计都了然于胸;作为一名导师,更是谆谆善诱、诲人不倦。正是在这样的导师的指导下,我才能在技术上和做事上有了较大的提高。在此,对朱老师衷心的说一声“谢谢”!

感谢朱维庆教授。朱维庆教授是实验室的创始人,正是有了老先生的指导,实验室多年来蒸蒸日上,为我们研究生的成长提供了优越的环境。当日选择来中科院读研究生,也是希望能目睹中科院老前辈们的工作风采,以受熏陶,如今愿望成真,老先生的工作热情鼓舞了我追求自己梦想的勇气。

感谢武岩波老师。初到实验室,不知所措,是他,让我慢慢适应环境,他的细心指点、自己的学习体会、学习规划以及未来研究方向,都热情地给我讲述,让我感慨自己知识的狭窄,惊喜于未来知识的广阔。在毕设开始以致结束整个过程,武岩波老师总是为我解开各种疑惑和问题,并对设计提出非常有意义的建议,在论文撰写的时候,他也给我提出很多结构上的问题,让论文条理清晰。再次感谢武岩波老师。

感谢水声通信网项目组的徐立军、傅翔、李欣国、魏振坤、孙兴涛、邢泽平等,感谢他们对我学习和生活上的指导和照顾,感谢他们陪伴我度过千岛湖实验那段美好的时光。

感谢王季煜师兄和李海莲师姐。他们对学业认真的态度,对问题冷静的分析方法,都是值得我学习的。他们不仅在学习上给予我诸多指点,在生活上也对我颇多照顾。

感谢我的同学崔兴隆、赵二亮、马驰、陈若婷,感谢我的师弟师妹许浩、张威、李丹丹、曹松军、樊艳强,陪伴我度过研究生时光。

最后,我要感谢的是我最亲爱的父母。在我二十多年的成长过程中,你们无时无刻无私地关怀和奉献,是我独在异乡求学的最大精神支柱,也是我可以依偎的最温馨港湾。你们是我永远的牵挂和眷恋!谨以此文献给我挚爱的双亲。

