卷积 Turbo 码的 RBF 神经网络译码算法

电路与系统

硕士生：　王一波

指导教师：　刘星成　副教授

摘　要

Turbo 码是一类新的纠错控制码，并已被确定为第 3 代移动通信系统的标准之一，如 CDMA-2000 和 WCDMA。自从 1993 年 C. Berrou 等人提出 Turbo 码后，大量的研究都集中在 Turbo 码的译码算法上。本文从考虑降低卷积 Turbo 码的译码复杂度，便于硬件实现的角度出发，提出卷积 Turbo 码的 RBF 神经网络译码算法，并讨论了三种译码算法。分别是：

(1) 卷积码的 RBF 神经网络译码算法。采用该算法的时候，引入了滑动窗口的方法，将卷积码译码和 RBF 神经网络结合，用并行处理的方式降低了译码时延。经过仿真可知，随着窗口的增大，这种译码算法的性能会越来越接近 Viterbi 软判决译码的性能。

(2) 没有外信息的卷积 Turbo 码的 RBF 神经网络译码算法。该方法在卷积码的 RBF 神经网络译码算法的基础上，将这种译码算法推广到卷积 Turbo 码的译码。仿真结果表明，所提出的算法能获得较好的译码性能。同时，译码复杂度相比 MAP 算法和 Max-Log-MAP 算法都有很大程度的降低。

(3) 带有外信息的卷积 Turbo 码的 RBF 神经网络译码算法。该方法在没有外信息的卷积 Turbo 码的 RBF 神经网络译码算法的基础上，进行了更深一步的研究，提取了外信息。仿真表明，所提出的算法在不提高复杂度的情况下，可以

获得更好的译码性能。

卷积 Turbo 码的 RBF 神经网络译码算法是一种新的译码算法。该算法具有并行处理能力，能够对卷积 Turbo 码进行有效的迭代译码，译码性能较好，运算复杂度相比 MAP 算法和 Max-Log-MAP 算法都有很大程度的降低。因此，该方法使得卷积 Turbo 码的译码算法更为简洁，易于硬件实现和实际系统中的应用。

关键词：

卷积 Turbo 码，迭代译码，径向基函数(RBF)，神经网络，误比特率(BER)

Decoding Algorithm of Convolutional Turbo Codes with RBF Neural Networks

Major : Circuit and System

Name: Wang Yi-bo

Supervisor: Associate Professor Liu Xing-cheng

Abstract

Turbo codes are a class of newly introduced error control codes adopted in communication proposals and standards, such as CDMA-2000 and WCDMA. The research of the codes has been focused on its decoding algorithms since its invention in 1993 by C. Berrou et al. In this thesis we proposed decoding algorithm of convolutional turbo codes with RBF neural networks. The algorithm could reduce complexity in decoding and be easily implemented with hardware. Three algorithms were discussed.

(1) Decoding algorithm of convolutional codes with RBF neural networks. The proposed algorithm could decode convolutional codes with time-sliding window and shorten time-delay. Simulation results show that the decoding performance with the proposed algorithm could be much more close to soft-decision Viterbi algorithm when the window size is increased.

(2) Not having exterior information decoding algorithm of convolutional turbo codes with RBF neural networks. The proposed algorithm was based on the decoding algorithm of convolutional codes with RBF neural networks. Simulation results show that the decoding performance with the proposed algorithm is better. Furthermore, it has the inherent advantages of higher speed in decoding process and less complicated

in computation than MAP algorithm and Max-Log-MAP algorithm.

(3) Having exterior information decoding algorithm of convolutional turbo codes with RBF neural networks. The proposed algorithm was based on the not having exterior information decoding algorithm of convolutional turbo codes with RBF neural networks and obtained exterior information. Simulation results show that the decoding performance with the proposed algorithm could be much better than the algorithm without exterior information. Furthermore, its complexity was not increased.

Decoding algorithm of convolutional turbo codes with RBF neural networks is a novel decoding algorithm for convolutional turbo codes. The algorithm has the parallel processing capability and is less complicated in computation than MAP algorithm and Max-Log-MAP algorithm. Its error correction performance is better. The algorithm could be easily implemented with hardware.

Key words:

Convolutional Turbo Code, Iterative Decoding, Radial Basis Function (RBF), Neural Networks (NN), Bit Error Rate (BER)

第 1 章 绪论

随着现代通信技术和计算机技术的迅速发展，每天都在不断涌现新的通信业务和信息业务，同时用户对通信质量和数据传输速率的要求也在不断提高。由于通信信道固有的噪声和衰落特性，信号在经过信道传输到达通信接受端的过程中不可避免地会收到干扰而造成信号失真。通常需要采用差错控制码来检测和纠正由信道失真引起的信息传输错误。由于差错控制码主要用于实现信道纠错，因此又称为纠错码或信道码。最早的差错控制码主要是用于深空通信和卫星通信，随着数字蜂窝电话、数字电视以及高分辨率数字存储设备的出现，编码技术的应用已经不仅仅局限于科研和军事领域，而是逐渐在各种实现信息交流和存储的设备中得到成功应用。

1.1 数字通信系统的结构

随着对高效、高可靠性数字通信系统需求的迅猛增长，大规模高速宽带网络的发展使语音、图像和其他多媒体信息的传输成为可能。通信系统设计人员最关心的是如何在数据源功率和传输带宽有限，系统复杂性和设备造价尽可能小的条件下实现尽可能准确的信息传输，即使信息传输的误码率最小化。信道编码是消除或降低信息传输错误概率的有效手段之一。下面首先介绍通用数字通信系统[1]的基本组成结构，如图 1-1 所示。

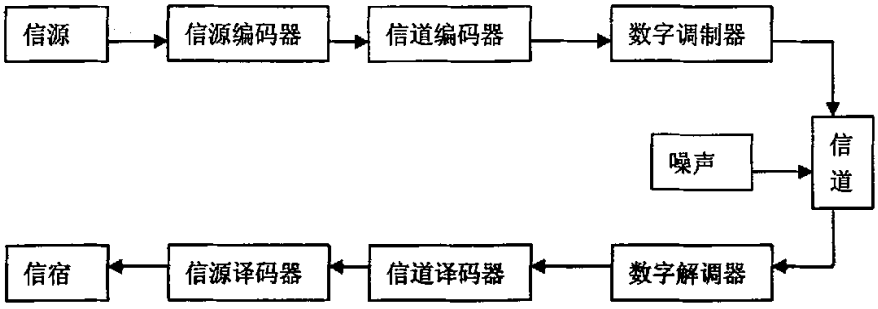


图 1-1 数字通信系统基本组成结构

图中，信源编码器是把信源发出的消息如语言、图像、文字等转换成为二进制(或多进制)形式的信息序列，并且为了使传输有效，还去掉了一些与传输信息无关的多余度。为了抗击传输过程中的各种干扰，往往要人为地增加一些多余度，使其具有自动检错或纠错能力，这种功能由图中的信道编码器即纠错编码器完成。调制器的功用是把纠错码送出的信息序列通过调制器变换成适合于信道传输的信号。数字信号在信道传输过程中，总会遇到各种干扰而使信号失真，这种失真信号传输到接收端的接收机，进行解调，变成二进制(或多进制)信息序列。由于信道干扰的影响，该信息序列中可能已有错误，经过信道译码器即纠错码译码器，对其中的错误进行纠正，再通过信源译码器恢复成原来的消息送给用户。

1.2 Turbo 码的发展及研究现状

1.2.1 Turbo 码编译码方案的提出

信息与通信系统中的编码有 4 种形式[2]: 信源编码、信道编码、密码编码和多址编码。信源编码解决了通信系统的有效性问题, 通过压缩信源冗余信息来提高通信的效率; 信道编码则是通过增加冗余位来达到保证通信系统的可靠性(通过牺牲带宽或传输速率来换取可靠性); 密码编码则是保证了系统的安全性; 多址编码主要是解决多用户通信问题。Shannon 第二编码定理证明, 用任意接近信道容量 C 的传输速率 R 传送，并且传输的差错率可以任意小的编码方法是存在的。信道编码的任务就是寻找这种编码。

根据 Shannon 有噪信道编码定理，在信道传输速率 R 不超过信道容量 C 的前提下，只有在码组长度无限的码集合中随机的选择编码码字并且在接收端采用最大似然译码算法时，才能使误码率接近为零。但是最大似然译码的复杂性随编码长度的增加而加大，当编码长度趋于无穷大时，最大似然译码是不可能实现的。所以人们认为随机性编译码仅仅是为证明定理存在性而引入的一种数学方法和手段，在实际的编码构造中是不可能实现的。

在 Turbo 码问世以前，前向差错控制码主要包括分组码、卷积码以及级联码等结构形式。虽然软判决译码、级联码和编码调制技术都对信道码的设计和发展产生了重大影响，但是其增益与 Shannon 理论极限始终都存在 2~3dB 的误差。因此在 Turbo 码提出以前，信道截止速率 R_0 一直被认为是差错控制码性能的实际极限，Shannon 极限仅仅是理论上的极限，是不可能达到的。

在 1993 年于瑞士日内瓦召开的国际通信会议(ICC'93)上，两位任教于法国不列颠通信大学的教授 C. Berrou、A. Glavieux 和他们的缅甸藉博士生首次提出了一种新型信道编码方案——Turbo 码[3][4]，由于很好的应用了 Shannon 信道编码定理中的随机性编译码条件，从而获得了几乎接近 Shannon 理论极限的译码性能，巧妙地将卷积码和随机交织器结合在一起，实现了随机编码的思想；同时，采用软输出迭代译码来逼近最大似然译码。计算机仿真结果表明：若采用大小为 65536 的随机交织器，并进行了 18 次迭代，则在 $E_b/N_0 \geq 0.7$dB 时，码率 R 为 1/2 的 Turbo 码在 AWGN 信道上的误比特率(BER)$\leq 10^{-5}$，达到了接近 Shannon 界的性能(码率 R 为 1/2 的 Shannon 界是 0dB)。

到目前为止，Turbo 码在现有信道编码方案中是很好的。Turbo 码一出现，就在编码理论界引起了轰动，成为自信息论提出一来最重大的研究进展。但由于几位发明者仅给出了一定参数下的计算机仿真结果，而没有严格的理论分析和解释，所以近些年在编码界掀起了 Turbo 码的研究热潮。Turbo 码论坛的成立以及于 1997 年 9 月和 2000 年 9 月在法国布莱斯特召开的两次 Turbo 码及相关主题国际会议都吸引了更多的编码界人士投身到 Turbo 码的研究中来，第三届 Turbo 码及相关主题国际会议也已经于 2003 年 9 月在法国布莱斯特召开。Turbo 的出现为编码理论和实践带来了一场革命，标志着长期将信道截止速率 R_0 作为实际容量限的历史结束，同时 Turbo 码还改变了人们设计信道码的传统观点，使信道编

码理论与技术的研究进入了一个崭新的阶段。

尽管目前对 Turbo 码的作用机制尚不十分清楚，对迭代译码算法的性能也还缺乏有效的理论解释，但它无疑为最终达到 Shannon 信道容量开辟了一条新的途径，其原理思想在相关研究领域中具有广阔的应用前景。目前，Turbo 码被看作 1982 年 TCM 技术问世以来，信道编码理论与技术研究上所取得的最伟大的技术成就，具有里程碑的意义。

1.2.2 Turbo 码的研究状况

在 Turbo 码的设计和理论研究方面，首先进行了 Turbo 码的结构研究。组成 Turbo 码的子码的选择，可以是典型的卷积码，也可以是分组码[5][6]。Turbo 码的结构可以是并行级联，也可以是串行级联，还可以是混合结构。交织器的设计对 Turbo 码至关重要，开始的研究主要注意的是交织器对 Turbo 码的 BER 性能和时延的影响。随着 Turbo 码逐步走向应用，人们注意到改进交织器的结构可以使 Turbo 码译码并行进行[7]，即使应用双精度的 SOVA 译码算法性能也不会降低很多——非常接近 Log-MAP 译码的性能。国内在交织器设计方面也有较多的研究，如王育民等研究组设计的复合型交织器[8]。在 Turbo 码距离谱的研究方面，Podemski 和 Daneshgaran 等人提出了计算汉明距离谱的算法，Perez 则通过距离谱解释了 Turbo 码性能曲线在大信噪比时变平坦——差错平底的现象。I. Sason 和 T. M. Duman 等人分别对交织衰落信道和高速 Turbo 码的性能限作了研究[9]。

在译码算法研究方面，主要时在最大后验概率(MAP)算法[10]和软输出 Viterbi 算法(SOVA)的基础上寻求新的算法和对原有的算法进行改进，降低译码复杂度。Turbo 码的译码算法主要有上述两种迭代算法(MAP 和 SOVA)。这方面的研究特别活跃，国内外有大量的研究论文涌现。例如：王新梅等对级联系统的软输出译码算法作了研究[11]。吴伟陵等对 Ralyeigh 衰落信道进行了分析[12]，提出了 Rake 接收，最大比合并、除以信道幅度参数的平方之和，以此为基础对以 Turbo 译码的算法进行改进，结果表明纠错性能得到很大提高。李建东等针对无线 OFDM 系统中的频率选择性衰落信道进行了 SOVA 译码算法研究[13]，提出了盲信道估计的 SOVA 和差分 SOVA 两种译码算法。王东明等人[14]将基于

MMSE 的 Turbo 检测译码方案应用于空时分组码分块传输系统，利用信道矩阵的循环特性，Turbo 信道检测算法可以用 FFT/IFFT 快速实现，并且矩阵求逆运算量非常低。采用 Turbo 码的 STBC-CP-SCBT（空时分组编码-加循环前缀的单载波分块传输）系统经过 6 次迭代检测后可以得到 2dB 的增益[14]。研究译码算法的同时，研究人员对译码的收敛性也没有忽视。Mihaljevic 和 Golic 对概率译码的收敛性进行了分析，这种分析对减少译码时延起了积极作用。Michael Tuchler[15]的研究则表明，根据简化的外部信息转换图，可以优化级联系统译码的收敛性，且在有限次迭代后的性能非常好。

Turbo 码的应用研究主要集中在如下一些方面：Turbo 编码调制、Turbo 信道检测和均衡、Turbo 码与 OFDM、Turbo 码与 MIMO 系统的结合改善系统纠错性能和提高系统容量，抵抗符号间干扰等等。例如，以 Robertson、Divsalar 等人为代表进行的网格编码调制（TCM）和 Turbo 编码相结合的所谓 T-TCM 联合编码调制研究；以 Hanzo 等人为代表的基于 Turbo 码的多用户检测、Turbo 均衡和干扰对消等方面的研究；以及 K. Mohamed-Pour 等人提出的在 Turbo 码迭代译码系统中使用判决反馈均衡等等，这些都属于应用基础研究[16][17][18][19][20][21][22]。国内在这方面的研究也非常活跃。姚彦等人的自适应编码调制方案可以获得较大的 SNR 增益，利用 Turbo 编码调制在 OFDM 子载波上的分机能力有效对抗信道时变带来的吞吐性能恶化[23]。宋文涛等人提出的 Turbo 编码多元调制方案是一种功率和频谱高效的编码调制方式，它比传统的网格编码调制（TCM）方式有更好的性能[24]。对于低复杂性的 Turbo 编码调制技术，刘增基等人也证实了它具有高频带利用率的特征，而且译码无需先验信道状态信息，在衰落因子 $f_D T_s$ 不大的平坦衰落信道中，基于 QPSK 等调制的 DD-TPCM 系统在 10^{-5} 误比特率时都没有差错平底，这是迄今为止所取得的相当好的结果[25]。在 Turbo 码与空时码、OFDM 系统、MIMO 系统以及分集接收技术中的应用，也有大量的研究成果报道[22][23][26]。在 Turbo 码的混合纠错方面，利用 Turbo 码的编码和译码特点进行混合 ARQ 设计，可以提高系统的整体性能[27]。

1.3 人工神经网络的特性

1.3.1 人工神经网络简介

人工神经网络(ANN)，亦称为神经网络(NN)，是由大量处理单元(神经元Neurons)广泛互连而成的网络，是对人脑的抽象、简化和模拟，反映人脑的基本特征。人工神经网络的研究是从人脑的生理结构出发来研究人的智能行为，模拟人脑信息处理的功能。

神经网络也经常被称为神经计算机，但它与现代数字计算机的不同之处主要表现在以下方面：

(1) 神经网络的信息存储与处理(计算)是合二为一的，即信息的存储体现在神经元互连的分布上；传统计算机的存储与计算机是独立的，因而在存储与计算之间存在着瓶颈。

(2) 神经网络具有很强的鲁棒性和容错性，善于联想、概括、类比和推广，任何局部的损伤不会影响整体结果。

(3) 神经网络具有很强的自学习能力，能为新的输入产生合理的输出，可在学习过程中不断完善自己，具有创新特点。

(4) 神经网络是一大规模自适应非线性动力系统，具有集体运算的能力。这与本质上是线性系统的现代数字计算机迥然不同。

人工神经网络是近年来的热点研究领域，涉及到电子科学与技术、信息与通信工程、计算机科学与技术、电气工程、控制科学与技术等诸多学科，其应用领域包括：建模、时间序列分析、模式识别和控制等，并在不断的拓展。

1.3.2 人工神经网络的信息处理能力

人工神经网络的计算机能力有三个显著的特点：一是它的非线性特性；二是大量的并行分布结构；三是它的学习和归纳能力。归纳指神经网络在学习(训练)过程中能为新的输入产生合理的输出。具有了这些特性的人工神经网络能够解决许多复杂的问题。

按照 Simon Haykin 的观点，人工神经网络具有如下的信息处理能力：

(1) 非线性。一个人工神经元可以是线性或非线性的。

(2) 输入一输出映射。人工神经网络具有学习能力，通过学习，人工神经网

络具有很好的输入－输出映射能力。一个流行的学习范例称为有导师学习或监督学习，它利用一组被标记的训练样本或任务实例对神经网络的突触权值进行调整。每一组训练样本包含一个给定的输入信号和对应的期望输出。对应给定的输入信号，网络产生实际的输出，实际输出与期望输出之间存在误差，通过某些规则反复修正网络的突触权值，可使实际输出与期望输出之间的误差减小到满意的程度，这样就完成了输入到输出的映射。

(3) 适应性。神经网络具有调整突触权值以适应周围环境的变化能力，尤其在特定环境中训练过的神经网络能很容易地被再次训练以处理环境条件微小的变化，这反映了神经网络的适应性。

(4) 容错性。容错包括空间上的容错、时间上的容错和故障检测。

(5) 超大规模集成的可执行能力。神经网络所具有的大量并行特性使其对特定任务的计算变得很快，这种特性也使神经网络非常适合超大规模集成技术的执行。

1.3.3 神经网络译码的研究现状

在利用神经网络对纠错码译码的研究方面[28][29][30][31]，在 1990 年，Caid Willian R.和 Means Robert W.利用 BP（误差反向传播算法）神经网络对 Hamming (7, 4) 分组码和 (2, 1, 2) 系统和非系统卷积码进行译码[28]；1994 年，E. El-Khamy Said E., Youssef El-Sayed A.和 Abdou Hossam-El-Din M.利用 BP 神经网络对分组码进行软判决译码[29]；1999 年，Annauth R.和 Rughooputh Harry C. S 利用 BP 神经网络对卷积 Turbo 码进行译码[31]。虽然已经将神经网络应用到纠错码译码方面，但是译码性能相比较一些传统的算法差，在降低误码率时不够理想。

1.4 本文研究意义和创新点

1.4.1 研究意义

由于 Turbo 码具有接近 Shannon 理论的性能，尤其是低信噪比下的优异性能使 Turbo 码在许多通讯系统都有非常大的应用潜力。除了在深空通信、卫星通

信以及多媒体通信等领域应用外，Turbo 码在无线移动通信系统中的应用是目前研究的热点[32][33][34]。C. Schurgers 研究了通过自适应调整迭代次数实现室内无线环境下 Turbo 码迭代译码的方案。T. Keller 和 J. P. Woodard 等人考察了基于 Turbo 码的并行 Modem 在个人通信中的应用问题。目前，Turbo 码已经成为第 3 代通信系统的标准之一，有关其关键技术的标准化也已经出现。在第 3 代移动通信系统 IMT-2000 中，由于移动信道传输媒质的不稳定性和噪声的不确定性，一般的纠错码很难达到较高要求的业务质量，而 Turbo 码因其具有独特的优势，已经被确定为 CDMA2000 辅助业务信道中高质量、高速率传输业务的可选编码方案。CDMA2000 前向和反向业务信道编码中，采用了码率为 1/2、1/3、1/4 的 Turbo 码。在 CDMA 移动通信系统中，由于系统容量(每个小区的最大用户数)与信噪比(SNR)近似成反比，则对于给定的误比特率(BER)，若降低 SNR，则系统容量增大。所用卷积码的约束长度一般都比较大(k=9)，这样使设备复杂度和成本都上升。而 Turbo 码能提供比一般的纠错码多几 dB 的编码增益。使用 Turbo 码在达到同样效果时，可降低设备复杂度和成本。故在 CDMA 系统中采用 Turbo 码技术可进一步提高容量。

Turbo 码还可以推广到 CDMA 多用户检测中，实现基于 Turbo 码译码原理的 CDMA 多用户检测接收机。在 CDMA 系统中，大量用户在同一传输媒质的同一频段、同一时间传输，而且各用户分配的编码波形不是正交的，另外由于实际条件的限制(如有限传输带宽、用户数、功率、信道、同步等)，使得在接收机中存在严重的多址干扰。多址干扰主要是由用户间扩频序列的相关性造成，而 Turbo 码在其编码系统中采用交织器，从而可以通过分散信息码元的位置来降低扩频码间的相关性，最终达到提高系统性能的目的。在实际的应用中，把 Turbo 码与 DS-CDMA 系统的扩频编码结合起来，Turbo 码编码作为外码，扩频编码作为内码，其间用交织器级联起来，类似于级联码的形式。在接收端，先通过匹配滤波器分离出各个用户的接收信息，然后根据信道模型计算传递条件概率并进行解扩，经过分支概率产生器后得到各个用户接收信息的后验概率，分别送入相应的 Turbo 码译码器。每个 Turbo 码译码器得到一个软判决输出和一个外部信息。其中外部信息送回到分支概率产生器，作为下一次迭代译码的先验信息，从而实现 Turbo 码的迭代译码。此外，Turbo 码在文本传输和数据存储等方面也有

应用。

由于 Turbo 码以前的各种译码算法都存在算法复杂度与它们可获得的性能及可实现性之间的矛盾，因此，目前国内外在 Turbo 码的译码算法实现上没有形成定论。

选择译码用神经网络算法来代替以前的各种译码算法，也正是基于上述原因的考虑。神经网络算法的主要问题是训练网络的权值，在训练网络权值方面所需要的计算量和时间相对较多，一旦网络的权值能够确定，在译码的时候只是将所接收到的码字和权值进行加减运算就可以得到输出。通过对 RBF 神经网络译码算法的仿真实验，相比较以前的译码算法而言，在利用 RBF 神经网络算法译码时所用到的计算量会减少许多，运算复杂度降低很多。在硬件实现方面，由于输出的计算方法会比较简单，这样比较易于硬件电路的设计和实现，使 Turbo 码在现实中较为方便的应用。

1.4.2 本文创新点

在目前所掌握的文献中，还没有关于利用 RBF 神经网络对卷积 Turbo 码进行译码的文章。可以说，在这个领域，还没有学者进行研究。RBF 神经网络结构简单、训练简洁而且学习收敛速度快，能够逼近任意非线性函数。同时，RBF 神经网络使用局部指数衰减的非线性函数(如高斯函数)对非线性输入输出映射进行局部逼近。这就意味着，逼近非线性输入输出映射，要达到相同的精度，RBF 神经网络所需要的参数要更少。通过对 RBF 神经网络特点的研究，提出卷积 Turbo 码的 RBF 神经网络译码算法，提出两种算法——无外信息和有外信息，并用 Matlab 编程对译码性能进行了仿真，取得了比较好的效果。具体算法及性能仿真详见第 3 章。

1.4.3 文章内容安排

第 1 章 即本章，介绍移动通信的发展简史，简述目前 3G 通信的关键技术之一——Turbo 码，介绍国内外一些卷积 Turbo 码译码的研究成果。同时，阐述论文选题意义，简要介绍文章的内容安排。

第 2 章 介绍本文涉及的信道编码基础知识和 Turbo 码的编译码原理。

第 3 章 介绍 RBF 神经网络译码算法，并对提出的卷积码的 RBF 神经网络算法、卷积 Turbo 码的 RBF 神经网络译码算法，利用 Matlab 编程进行性能仿真，并对不同算法进行比较和分析。

第 4 章 对研究结果进行分析和总结，提出了有待进一步研究的问题。

第 2 章 Turbo 码编译码原理

2.1 基础知识介绍

(1) 随机错误：由随机噪声引起的错误，由于随机噪声的特性，该错误的特点是各码元是否发生错误是互相独立的，通常不会成片地出现错误。

(2) 突发错误：由突发噪声引起的错误。由于突发噪声的特性，使“各个码元是否错误”存在相关性。因此，该错误是成片出现的。在一个突发错误持续长度内，开头和最末的码元总是错的，中间的一些码元则可以出错也可以不出错，单从统计角度看，中间的码元出错的几率较高。

(3) 高斯正态分布：

高斯正态分布的表达式和高斯概率密度函数曲线如图 2-1 所示。其中：σ 表示标准差，μ 为均值。高斯分布以均值对称，峰值处于均值处，最小值在 $\pm\infty$ 处，形状如同“钟形”。

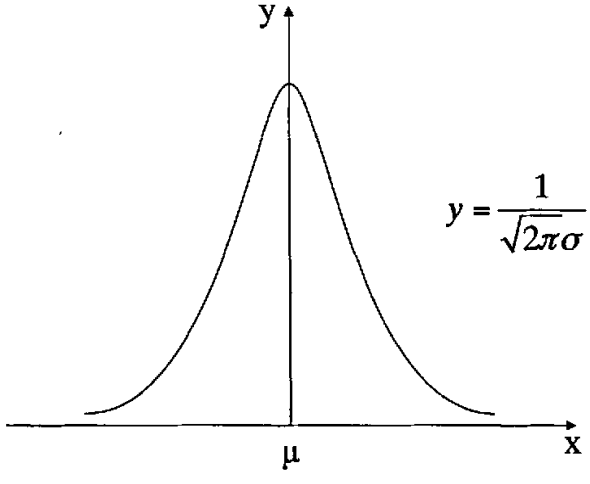


图 2-1　高斯正态分布

2.2 最大似然译码

译码器的基本任务就是根据一套译码规则，由接收序列 R（经过噪声信道）给出与发送的信息序列 M 最接近(最好是相同)的估值序列 M'。由于 M 与码字 C 之间存在一一对应关系，所以这等价于译码器根据 R 产生一个 C 的估值序列 C'。显然，当且仅当 $C'=C$ 时，$M'=M$，这时译码器正确译码。

如果译码器输出的 $C'\neq C$，则译码器产生了错误译码。之所以产生错误译码是由于：信道干扰很严重，超过了码本身的纠错能力；其次，由于译码设备的故障。当给定接收序列 R 时，译码器的条件译码错误概率定义为

$$P(E|R)=P(C'\neq C|R)。\tag{2-1}$$

所以译码器的错误译码概率

$$P_E=\sum_R P(E|R)P(R),\tag{2-2}$$

$P(R)$是接收 R 的概率，与译码方法无关，所以译码错误概率最小的最佳译码规则是使

$$\min P(C'\neq C|R)\Rightarrow \max P(C'=C|R),\tag{2-3}$$

因此，如果译码器对输入的 R，能在 2^k 个码字中选择一个使 $P(C'=C|R)$ $(i=1,2,\cdots,2^k)$ 最大的码字 C_i 作为 C 的估计序列 C'，则这种译码规则一定使译码器输出错误概率最小，称这种译码规则为最大后验概率译码。

由贝叶斯公式

$$P(C_i|R)=\frac{P(C_i)P(R|C_i)}{P(R)},\tag{2-4}$$

可知，若发送端发送每个码字的概率 $P(C_i)$均相同，且由于 $P(R)$与译码方法无关，所以

$$\max_{i=1,2,\cdots,2^k} P(C_i|R)\Rightarrow \max_{i=1,2,\cdots,2^k} P(R|C_i)。\tag{2-5}$$

对于 DMC 而言

$$P(R|C_i)=\prod_{j=1}^{n} P(r_i|c_{ij}),\tag{2-6}$$

这里码字 $C_i=(c_{i1},c_{i2},\cdots,c_{in})$，$i=1,2,\cdots,2^k$。

一个译码器的译码规则若能在 2^k 个码字 C 中选择某一个 C_i 使式(2-5)成为最

大，则这种译码规则称为最大似然译码(MLD)，$P(R|C)$称为似然函数，相应的译码器称为最大似然译码器。由于$\log_b x$与x是单调关系，因此式(2-5)与(2-6)可写成

$$\max_{i=1,2,\cdots,2^k} \log_b P(R|C_i) = \max_{i=1,2,\cdots,2^k} \sum_{j=1}^{n} \log_b P(r_i|c_{ij}), \qquad (2\text{-}7)$$

称$\log_b P(R|C)$为对数似然函数或似然函数。对于 DMC 信道，MLD 是使译码器错误概率最小的一种最佳译码准则或方法，但此时要求发端发送每一码字的概率$P(R|C_i)$ $(i=1,2,\cdots,2^k)$均相等，否则 MLD 不是最佳的。

2.3 卷积码

卷积码是 1955 年由 Elias 提出的，它与分组码不同。分组码编码时，本组中的$n-k$个校验元与本组的k个信息元有关，而与其它各组码元无关。分组码译码时，也仅从本码组中的码元内提取有关译码信息，而与其它各组无关。卷积码(n_0, k_0, m)，其中n_0为码长，k_0为信息位，m为编码存贮。在其编码时，本组的n_0-k_0个校验元不仅与本组的k_0个信息元有关，而且还与以前各时刻输入至编码器的信息组有关。同样在卷积码译码过程中，不仅从此时刻收到的码组中提取译码信息，而且还要利用以前或以后各时刻收到的码组中提取有关信息。此外，卷积码中每组的信息位k_0和码长n_0，通常也比分组码的k和n要小。

正由于在卷积码的编码过程中，充分利用了各组之间的相关性，且k_0和n_0也较小，因此，在与分组码同样的码率R和设备复杂的条件下，无论从理论上还是从实际上均已证明卷积码的性能至少不比分组码差，且实现最佳和准最佳译码也较分组码容易。所以，从信道编码定理看，卷积码是一种非常有前途的，能达到信道编码定理所提出的码类。但由于卷积码各组之间相互有关，因此在卷积码分析过程中，至今仍未找到像分组码那样有效的数据工具，以致性能分析比较困难，从分析上得到的成果也不像分组码那样多，而往往还要借助计算机的搜寻来找寻好码。

但由于卷积码各组的n_0、k_0均比分组码小，译码似乎比分组码要容易，并且卷积码有三种比较好的译码方法：(1)1963 年由 Massey 提出的门限译码，这是

一种利用码代数结构的代数译码，类似与分组码中的大数逻辑译码；(2)1961 年由 Wozencraft 提出，1963 年由 Fano 改进的序列译码，这是基于码树图结构上的一种准最佳概率译码；(3)1967 年由 Viterbi 算法，这是基于码的网(trellis)图基础上的一种最大似然译码算法，是一种最佳的概率译码方法。

2.4 Turbo 码编译码原理

C. E. Shannon 在其“通信的数学理论”一文中提出并证明了著名的有噪信道编码定理，他在证明信息速率达到信道容量可实现无差错传输时引用了 3 个基本条件：

(1) 采用随机性编码。

(2) 编码长度 $L \to \infty$，即分组的码组长度无限。

(3) 译码过程采用最佳的最大似然译码(MLD)方案。

在信道编码的研究与发展过程中，基本上是以后两个条件为主要方向的。而对于条件(1)，虽然在码集合中随机选择编码码字可以使获得好码的概率增大，但是最大似然译码器的复杂性随码字数目的增加而加大，当编码长度很大时，译码几乎不可能实现。所以人们认为条件(1)仅仅是为证明定理存在性而引入的一种数学方法，在实际的编码构造中是不能实现的。事实上，分组码和卷积码都具有非常规则的结构，因此它们的编码器和译码器在一定的复杂性条件下是可实现的。但同时这种规则的编译码结构也使这些编码方法的性能与 Shannon 理论极限存在一定的差距。

Turbo 码通过在编码器中引入随机交织器，使码字具有近似随机的特性；通过分量码的并行级联实现了通过短码(分量码)构造长码(Turbo 码)的方法；在接收端虽然采用了次最优的迭代算法，但分量码采用的最优的最大后验概率译码算法，同时通过迭代过程可使译码接近最大似然译码。综合上述分析可见，Turbo 码充分考虑了 Shannon 信道编码定理证明时所假设的条件，从而获得了接近 Shannon 理论极限的性能。Turbo 码同时也第一次从实践中证明了信道编码定理的正确性。

2.4.1 Turbo 码的编码结构

Turbo 码的最大特点在于它通过在编译码器中交织器和解交织器的使用，有效地实现了随机性编译码的思想，通过短码的有效结合实现长码，达到了接近 Shannon 理论极限的性能。

C. Berrou 等人最初提出的 Turbo 码采用的是并行级联卷积码的结构。图 2-2 给出了由两个分量编码器组成的 Turbo 码的编码框图。

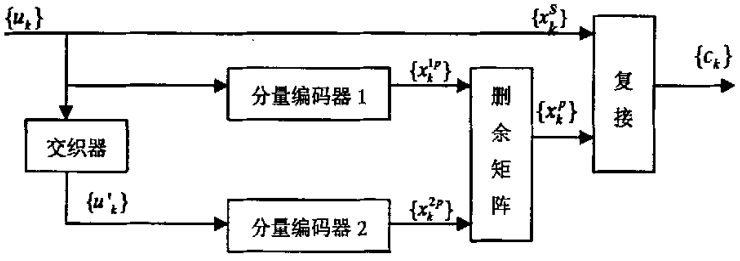


图 2-2 Turbo 码的编码结构

Turbo 码编码器[35]主要由分量编码器、交织器以及删余矩阵和复接器组成。分量码一般选择为的递归系统卷积码，当然也可以是分组码、非递归卷积码以及非系统卷积码。通常两个分量码采用相同的生成矩阵，当然，分量码也可以是不同的。

在 Turbo 编码过程中，两个分量码的输入信息序列是相同的，长度为 N 的信息序列 $\{u_k\}$ 在送入第一个分量编码器进行编码的同时作为系统输出 $\{x_k^s\}$ 直接送至复接器，同时 $\{u_k\}$ 经过交织器 I 后的交织序列 $\{u'_k\}$ 送入第二个分量编码器。两个分量编码器输入序列仅仅是码元的输入顺序不同。两个分量编码器输出的校验序列分别为 $\{x_k^{1p}\}$ 和 $\{x_k^{2p}\}$。为提高码率和系统频谱效率，可以将两个校验序列经过删余矩阵删余后（得到 $\{x_k^p\}$）再与系统输出 $\{x_k^s\}$ 一起经过复接构成码字序列 $\{c_k\}$。交织器和分量码的结合可以确保 Turbo 码编码输出码字都具有较高的汉明重量。Turbo 编码器中交织器的作用是将信息序列中的比特顺序重置。当信息序列经过第一个分量编码器编码后输出的码字重量较低时，交织器可使交织后的信息序列经过第二个分量编码器编码后以很大的概率输出高重码字，从而提高码字

的汉明重量；同时好的交织器还可以有效地降低校验序列间的相关性。通过交织，编码序列在长为 $2N$ 或 $3N$（不经过删余）比特的范围内具有无记忆性，从而由简单短码构造成了近似随机长码。因此，交织器设计的好坏在很大程度上影响着 Turbo 码的性能。删余矩阵的作用是提高编码码率。

2.4.2 Turbo 码的译码结构

Turbo 码获得优异性能的根本原因之一是采用了迭代译码，通过分量译码器之间软信息的交换来提高译码性能。对于 Turbo 码这样的并行级联码，如果分量译码器的输出为硬判决，则不可能实现分量译码器之间软信息的交换。从信息论的角度来看，任何硬判决都会损失部分信息，因此，如果分量译码器能够提供一个反映其输出可靠性的软输出，则其它分量译码器也可以采用软判决译码，从而系统的性能可以得到进一步提高。为此，人们又提出了软输出译码的概念和方法，即译码器的输入输出均为软信息。

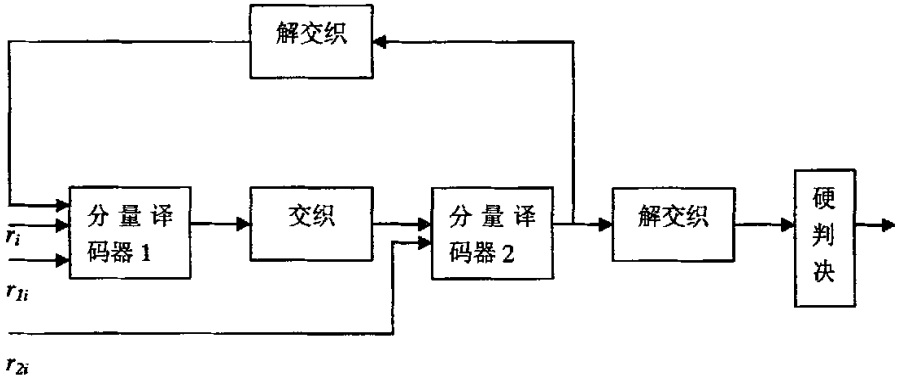


图 2-3　Turbo 码的译码结构

图 2-3 为 Turbo 码的译码结构。译码器接收端收到的经过噪声信道后的信息序列为 r_i，分量编码器 1 和分量编码器 2 产生的校验序列经过噪声信道后送到译码器的序列分别为 r_{1i} 和 r_{2i}。收到的信息序列 r_i 直接送给分量译码器 1，校验序列 r_{1i} 和 r_{2i} 分别送入分量译码器 1 和分量译码器 2。分量译码器 1 通过利用收到的信息序列和校验序列得到软判决的译码输出。这个软判决的译码输出经过交织器后送入分量译码器 2。分量译码器 2 通过利用所收到的校验序列和由分量译码

器 1 产生的软信息得到软判决的译码输出。这个软判决的译码输出经过解交织器反馈至分量译码器 1。分量译码器 1 再一次的产生软判决的译码输出，经过交织器后反馈给分量译码器 2，这样就实现了迭代。随着迭代次数的增加，两个分量译码器得到的外部信息值对译码性能提高的作用越来越少，在达到一定迭代次数后，译码性能不再提高。这时根据分量译码器 2 的输出对数似然比，经过解交织后再进行硬判决即得到译码输出。

2.4.3 软输出迭代译码算法

Turbo 码的编码过程实际上是一个利用强约束短码构造伪随机长码的过程，对于传统编码方法来说，当码字达到一定长度以后，MLD（最大似然译码）是不可能实现的。软判决迭代译码很好的解决了这个问题。Turbo 码通过在分量译码器之间交换外部信息来提高性能。Turbo 码的译码算法主要有两大类。一类是基于最大后验概率（MAP）的软输入算法，主要包括标准的 MAP 算法，对数域上的 Log-MAP 算法和 Max-Log-MAP 算法、修正的 MAP 算法（M-MAP）、滑动窗 MAP（SW-MAP）算法和只有向前推进的 MAP 算法（OSA）。其中 Log-MAP 算法是 MAP 算法的对数形式，它通过大量的乘法运算转化为加法运算来简化算法的复杂性。M-MAP 算法则是在 MAP 算法的基础上通过减少格图搜索状态达到简化的目的。通过对 Log-MAP 算法中分支路径度量算法的简化，就得到了 Max-Log-MAP 算法。另一类 Turbo 码算法是基于 Viterbi 算法（B-SOVA）和 List－SOVA 等。对传统 Viterbi 算法进行修正是因为它没有提供 Turbo 译码所要求的后验概率（APP）信息。此外，研究表明，Max-Log-MAP 算法与 B-SOVA 从硬判决角度看是等价的。也有一些对 Turbo 码迭代译码算法性能比较的讨论。

考虑到 Turbo 码译码的延时，S. Benedetto 等提出了面向流的连续 Turbo 码迭代译码算法；Y. ChangT 提出了并行 Turbo 码的方法；W. Yufei 通过后向推概率的前向计算来降低延时。其他有关迭代译码算法的讨论还有很多。考虑到 Turbo 码在未来高速数据传输中的应用，寻找低延时和低复杂性的软判决译码算法始终是 Turbo 码研究的重要课题。

2.4.3.1 MAP 译码算法

在 MAP 算法提出后的将近 20 年时间里，由于其大计算量和硬件实现高复杂性而一直没有得到重视。直到 1993 年 Turbo 码的发明者在最初的 Turbo 码迭代译码方案中采用了修正的 MAP 算法。相应的各种简化算法也不断出现。MAP 算法是基于码字格图的软输出译码算法，目的是使比特错误概率最小。根据 MLD 原理，译码器的主要任务就是计算在接收采样条件下不同发送符号的概率，而后将接收采样判决为概率值最大的信息符号。

图 2-4 所示的软输入软输出(SISO)译码器，它能为每一译码比特提供对数似然比输出。

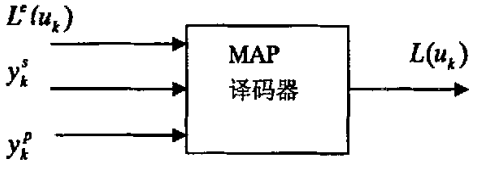


图 2-4 软输入软输出译码器框图

在图 2-4 中，MAP 译码器的输入序列为 $y = y_1^N = (y_1, y_2, \cdots, y_k, \cdots, y_N)$。其中，$y_k = (y_k^s, y_k^p)$。$L^e(u_k)$是关于$u_k$的先验信息，$L(u_k)$是关于$u_k$的对数似然比。她们的定义如下：

$$L^e(u_k) \triangleq \ln \frac{P(u_k = 1)}{P(u_k = 0)}, \tag{2-8}$$

$$L(u_k) \triangleq \ln \frac{P(u_k = 1 \mid y_1^N)}{P(u_k = 0 \mid y_1^N)}。 \tag{2-9}$$

假定发送端 RSC 编码器的存储级数为v，约束长度为K，编码器在k时刻的状态为$S_k = (a_k, a_{k-1}, \cdots, a_{k-v+1})$，编码输出序列$x_k = (x^s, x^p)$。传输信道模型如图 2-5 所示，从图中可知：

$$y_k^s = a_k^s c_k^s + n_k^s = a_k^s (2x_k^s - 1)\sqrt{E_s} + n_k^s, \tag{2-10}$$

$$y_k^p = a_k^p c_k^p + n_k^p = a_k^p (2x_k^p - 1)\sqrt{E_s} + n_k^p, \tag{2-11}$$

式中，a_k^s和a_k^p为信道衰落因子，对于 AWGN 信道，$a_k^s = a_k^p = 1$；n_k^s和n_k^p式两个

独立同分布的高斯噪声样值，它们的均值为 0，方差$\sigma^2 = N_0/2$。

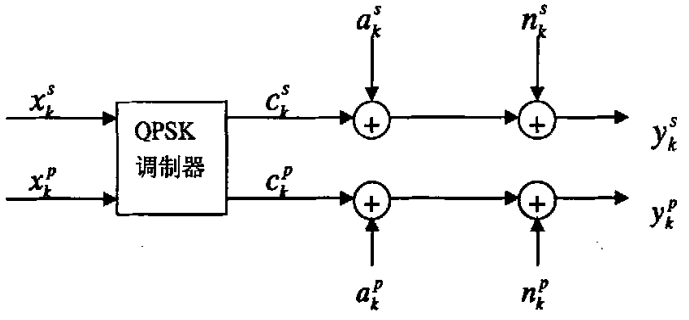


图 2-5　　信道模型

MAP 译码器的任务就是求解式(2-10)，然后按照下列规则进行判决

$$\hat{u}_k = \begin{cases} 1 & L(u_k) \geq 0 \\ 0 & L(u_k) < 0 \end{cases}。 \tag{2-12}$$

利用 BCJR 算法对式(2-9)的计算方法进行推导可得

$$\begin{aligned} L(u_k) &= \ln \frac{P(u_k = 1, y_1^N)/p(y_1^N)}{P(u_k = 0, y_1^N)/p(y_1^N)} \\ &= \ln \frac{\sum_{\substack{(s',s) \\ u_k=1}} p(S_{k-1} = s', S_k = s, y_1^N)/p(y_1^N)}{\sum_{\substack{(s',s) \\ u_k=0}} p(S_{k-1} = s', S_k = s, y_1^N)/p(y_1^N)} \\ &= \ln \frac{\sum_{\substack{(s',s) \\ u_k=1}} \tilde{\alpha}_{k-1}(s') \cdot \gamma_k(s',s) \cdot \tilde{\beta}_k(s)}{\sum_{\substack{(s',s) \\ u_k=0}} \tilde{\alpha}_{k-1}(s') \cdot \gamma_k(s',s) \cdot \tilde{\beta}_k(s)}, \end{aligned} \tag{2-13}$$

式中，求和是对所有由$u_k = 1$（或$u_k = 0$）引起的$S_{k-1} \to S_k$的状态转移进行的。其中各项计算方法如下：

$$\tilde{\alpha}_k(s) = \frac{\alpha_k(s)}{\sum_s \alpha_k(s)}, \quad \alpha_k(s) = p(S_k = s, y_1^k) = \sum_{s'} \alpha_{k-1}(s') \cdot \gamma_k(s',s); \tag{2-14}$$

$$\tilde{\beta}_k(s)=\frac{\beta_k(s)}{p(y_{k+1}^N\mid y_1^k)},\quad \beta_k(s)=p(y_{k+1}^N\mid S_k=s'),\quad \beta_{k-1}(s')=\sum_s \beta_k(s)\cdot\gamma_k(s',s);\ (2\text{-}15)$$

$$\gamma_k(s',s)=P(u_k)\cdot p(y_k\mid u_k),\qquad (2\text{-}16)$$

式(2-16)中，$P(u_k)$是u_k的先验概率，$p(y_k\mid u_k)$由信道转移概率决定。

2.4.3.2 SOVA 译码算法

对于卷积码，Viterbi 算法是最优的最大似然译码方法，译码输出为卷积码的最优估计序列。但对于属于级联卷积码的 Turbo 码而言，传统的 Viterbi 算法存在两个缺陷：首先，一个分量译码器输出中存在的突发错误会影响另一个分量译码器的译码性能，从而使级联码的性能下降。其次，无论是软判决 Viterbi 算法还是硬判决 Viterbi 算法，其译码输出均为硬判决信息，若一个分量码采用 Viterbi 算法译码，则另一个分量译码器只能以硬判决结果作为输入，无法实现软判决译码，从而性能会有所下降。因此，如果 Viterbi 译码器能够提供软信息输出，则可以弥补上述两个缺陷，并且可以通过在分量译码器之间软信息的交换使级联码的性能大大提高。为此，需要在传统的 Viterbi 算法上进行修正，使之提供软信息输出，相应的算法就成为软输出 Viterbi 算法，记做 SOVA (Soft Output Viterbi Algorithm)。

SOVA 算法是 Hagenauer 于 1989 年提出的。它是 Viterbi 算法的改进类型。Viterbi 算法是一种最大似然译码算法。通俗地来说，它的译码过程是在接收序列 R 的控制下，在码的篱笆图上走编码器走过的路径。

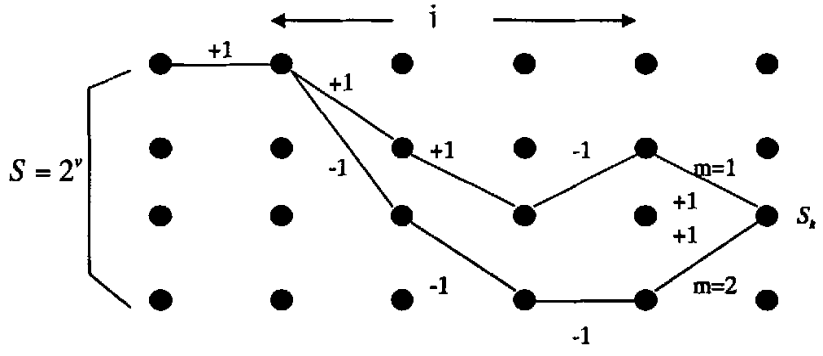


图 2-6　SOVA 算法的一个例子

图 2-6 表示的是一个 SOVA 算法的例子。为了简化起见，我们考虑网格图上每一个节点有两个分支，状态数为 2^v ，v 为编码器寄存器个数。它以 δ 为时延进行一比特判决，δ 足够大，使得 2^v 个幸存路径以足够大的概率汇聚于一点。在 k 时刻，对于状态 S_k，Viterbi 算法选择一条幸存路径，这是通过计算路径最小距离度量(或最大相关度量)而得到的。同时，状态 S_k 还对应一条待选路径。对于幸存路径，将其度量标为 M_1，相应的，待选路径的度量我们标为 M_2。于是幸存路径选错的概率为

$$p_{sk} = \frac{e^{-M_2}}{e^{-M_1}+e^{-M_2}} = \frac{1}{1+e^{M_2-M_1}} = \frac{1}{1+e^{\Delta}}, \qquad (2\text{-}17)$$

式中，$\Delta = M_2 - M_1 \geq 0$。$p_{sk}$ 代表的则是传输不可信度。于是在 e 个路径 1(幸存路径)与路径 2(待选路径)的信息比特不等的位置处，其错误概率为 p_{sk}。我们可以用下式表示：

$$\hat{p}_j \leftarrow \hat{p}_j(1-p_{sk}) + (1-\hat{p}_j)p_{sk} \quad j = j_1, \cdots, j_e(u_j^1 \neq u_j^2), \qquad (2\text{-}18)$$

式中，$\hat{p}_j$ 表示的是已存储的路径 1 的错误概率。则对数似然比可以写为

$$\hat{L}_j = \lg\frac{1-\hat{p}_j}{\hat{p}_j} \qquad 0 \leq \hat{L}_j < \infty, \qquad (2\text{-}19)$$

结合式(2-17)、式(2-18)和式(2-19)，可以得到

$$\hat{L}_j \leftarrow f(\hat{L}_j,\Delta)=\frac{1}{\alpha}\ln\frac{1+e^{(\alpha\hat{L}_j+\Delta)}}{e^{\Delta}+e^{\alpha\hat{L}_j}}, \tag{2-20}$$

式中，α 的引入是为了防止信噪比的增加而产生溢出。$\alpha=4d_{free}E_s/N_0$。上式可近似写为

$$f(\hat{L}_j,\Delta)=\min(\hat{L}_j,\Delta/\alpha)。 \tag{2-21}$$

于是 SOVA 算法可以分为以下几个步骤完成：

(1) 计算路径度量与度量差；

(2) 更新可靠性度量；

(3) 减去内信息，得到下一步所需的外信息值。

以上几步完成后，将所得到的外信息值带入下一个 SOVA 译码器中，进行下一步迭代，即可完 SOVA 算法在 Turbo 码中译码的应用。

SOVA 算法的运算量低于 MAP 算法，但其译码性能比 MAP 算法差。

2.4.4 迭代译码流程

Turbo 译码器的最大特点时采用了迭代译码。T. Richardson 从理论上对迭代译码过程进行了分析。S. ten Brink 利用分量译码器输出外部信息和输入采样的互信息定义了外部信息转移图，并利用此图实现对迭代译码过程的跟踪，从而估计迭代译码的收敛性。利用此图还可以预测实现一定性能要求时所必须的迭代译码次数。它为分析迭代译码过程和迭代译码方案的设计提供了有力的工具。

对于 Turbo 码来说，对所有数据帧都采用同样次数的译码迭代时没有必要的。有些数据帧经过很少几次迭代就可以实现无差错译码，而有些数据帧无论经过多少次迭代都不可能实现无差错译码。因此，为降低系统的实现复杂性和延时，动态选择译码迭代次数时非常必要的解决的办法就是根据迭代译码的特点设计相应的停止准则，当迭代译码满足这个准则时，就停止迭代，否则继续。有关 Turbo 迭代停止的准则研究已有很多：K. Gracie 提出了及早判决的迭代停止准则；S. Benedetto 提出了基于分量译码器输出软信息概率交叉熵最小化的停止准则；在此基础上 R. Y. Shao 提出了两个减小计算复杂性的停止准则——SCR 准则和 HAD 准则；W. Yufei 提出了基于分量译码器输出软信息硬判决符号变化的停止

准则；S. Kim 等人提出的停止准则以分量译码器输出外部信息的方差之差作为判据。A. Matache 对已有的基于软判决输出和硬判决输出的停止准则进行了综合分析比较。纵观已有的停止准则，基本上都是以连续迭代过程中分量译码器输出软判决信息的差别作为判决依据的。此外，还可以将 Turbo 码与 CRC 串行级联，以 CRC 校验来决定是否停止迭代(对每次迭代后译码器输出的硬判决结果进行 CRC 校验，如果 CRC 校验全部正确，则停止迭代，否则继续下一轮迭代)。

第 3 章 卷积 Turbo 码的 RBF 神经网络译码算法

3.1 RBF 神经网络

输入层、隐含层(一层或者多层)和输出层构成的神经网络称为多层前向神经网络。多层前向神经网络输入层中的每个源节点的激励模式(输入向量)单元组成了应用于第二层(如第一隐层)中神经元(计算节点)的输入信号，第二层输出信号成为第三层的输入，其余层类似。网络每一层的神经元只含有作为它们输入前一层的输出信号，网络输出层(终止层)神经元的输出信号组成了网络中输入层(起始层)源节点产生的激励模式的全部响应。即信号从输入层输入，经隐含层传给输出层，由输出层得到输出信号。

常见的两个多层前向网络[36]：BP 神经网络和 RBF 神经网络。

1985 年，Powell 提出了多变量插值的径向基函数方法。1988 年，Broomhead 和 Lowe 首先将 RBF 应用于神经网络设计，构成了径向基函数神经网络，即 RBF 神经网络。结构上看 RBF 神经网络属于多层前向神经网络。它是一种三层前向网络，输入层由信号源节点组成；第二层叫隐含层，隐单元的个数由所描述的问题而定，隐单元的变换函数是对中心点径向对称且衰减的非负非线性函数；第三层为输出层，它对输入模式做出响应。

RBF 神经网络的基本思想：用径向基函数(RBF)作为隐单元的“基”，构成隐含层空间，隐含层对输入矢量进行变换，将低维的模式输入数据变换到高维空间内，使得在低维空间内的线性不可分问题在高维空间内线性可分。它不仅没有 BP 神经网络所固有的局部极小值问题，而且其学习速度远远高于 BP 神经网络。RBF 神经网络结构简单、训练简洁而且学习收敛速度快，能够逼近任意非线性函数。同时，RBF 神经网络使用局部指数衰减的非线性函数(如高斯函数)对非线性输入输出映射进行局部逼近。这就意味着，逼近非线性输入输出映射，要达

到相同的精度，RBF 神经网络所需要的参数要更少。

RBF 网络是单隐层的前向网络，它由三层构成：第一层是输入层，第二层是隐含层，第三层是输出层。其结构如图 3-1 所示，其中，从输入层空间到隐含层空间的变换是非线性的，而隐含层空间到输出层空间的变换是线性的。

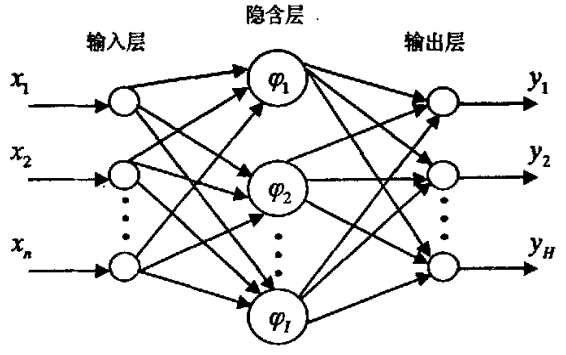


图 3-1　RBF 神经网络示意图

图 3-1 中隐含层共有 I 个中心 T_i（$i=1,2,3,\cdots I$），输出层共有 H 个神经单元（$j=1,2,3,\cdots H$），隐含层与输出层突触权值用 W_{ij} 表示，W_{ij} 表示第 i 个中心点连接到第 j 个输出节点的权值。RBF 神经网络的输入层有 n 个神经元，其中任意一个神经元用 m 表示；隐含层有 I 个神经元，任意一个神经元用 i 表示，第 i 个隐单元的激励输出为“基函数” $\varphi(X,T_i)$。若取高斯函数 G 为“基函数”，则可以表示为：

$$\varphi(X,T_i)=G(\|X-T_i\|)$$

$$=\exp\left(-\frac{\|X-T_i\|^2}{2\sigma_i^2}\right)=\exp\left(-\frac{\sum_{m=1}^{n}(x_m-T_{im})^2}{2\sigma_i^2}\right),\quad(3\text{-}1)$$

其中，$X=[x_1,x_2,\cdots,x_n]$ 为 RBF 网络的输入，$T_i=[T_{i1},T_{i2},\cdots,T_{im}]$ 为高斯函数的中心，σ_i 为高斯函数的方差。方差可用下式来计算：

$$\sigma_1=\sigma_2=\cdots=\sigma_I=d_{\max}/\sqrt{2I},\quad(3\text{-}2)$$

其中，I 为隐单元的个数，d_{max} 为所选取中心之间的最大距离。RBF 网络的输出层第 j 个神经元的输出按下式计算：

$$y_j = \sum_{i=1}^{I} W_{ij}\varphi\left(X, T_i\right), \tag{3-3}$$

其中，X 为网络的输入，最后得到的输出 $Y=[y_1, y_2, \cdots, y_H]$。

RBF 的中心选取有随机选取中心、自组织学习选取中心、正交最小二乘法选取中心等多种方法。本文根据各种编码样本数量选取最简单的随机选取中心法，这样既简化了编程，又在速度和结果上得到了满意的效果。

3.2 卷积码的 RBF 神经网络译码算法

由 3.1 的分析及 RBF 神经网络具有并行计算的能力，采用 RBF 神经网络译码算法来构成 Turbo 码的译码器。RBF 神经网络译码器代替图 2-3 中两个分量译码器。选择 (n_0, k_0, m) 系统卷积码编码器来代替图 2-2 中的两个分量编码器。这样对整个译码网络的训练，实际上转化为对单个分量译码器的训练。由于卷积 Turbo 码是由两个卷积码编码器构成，这样对卷积 Turbo 码的译码实际上转变为利用 RBF 神经网络对两个由卷积码编码器生成的卷积码进行译码。因此，我们先对基于 RBF 神经网络的卷积码译码进行研究。

3.2.1 卷积码的 RBF 神经网络译码器

卷积码 (n_0, k_0, m)，其中 n_0 为码长，k_0 为信息位，m 为编码存贮。在其编码时，本组的 n_0-k_0 个校验元不仅与本组的 k_0 个信息元有关，而且还与以前各时刻输入至编码器的信息组有关。

确定一个卷积码的方法有很多种，比如生成多项式、编码树图以及状态图等。这些方法都能唯一地确定出卷积码编码器的输入和输出关系。在用状态图表示的卷积码编码中，编码器的各种状态之间的转移确定了编码器的输入输出的过程。随着信息序列的不断送入，编码器就不断的从一个状态转移到另一状态，并输出相应的码序列，这样在状态图中就可以唯一地确定一条状态转移路径以及对应的编码输出序列。图 3-2 为码率 $R=1/2$，移存器数目 K=2 的卷积码编码器。

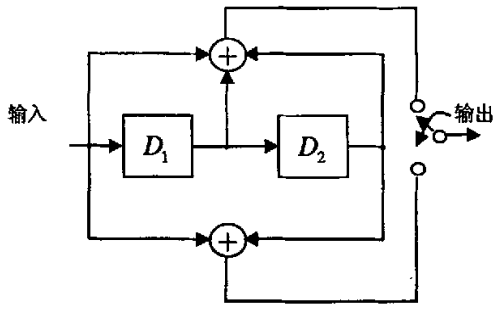
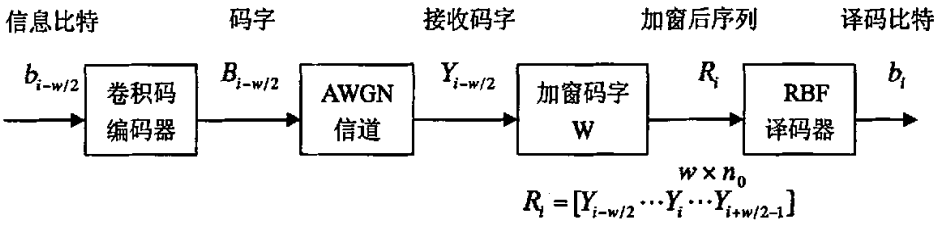


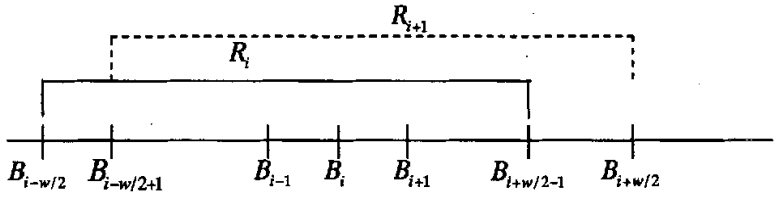
图 3-2　(2，1，2)卷积码编码器

在该例中，编码存贮$m=2$,$k_0=1$,编码器由两级移存器组成($K=2$)。因此，移存器中的存贮状态只有2^m种可能，也就是说状态的总数为2^m个。随着信息元的不断送入，编码器的状态在不断地转移，并输出相应的分支（子组），组成对应于输入信息序列的一个码序列。

应用于仿真的带有 RBF 神经网络译码器的通信系统的方框图如图 3-3(a)所示，其中所采用的 RBF 神经网络译码器的结构如图 3-4 所示，根据滑动分组方法[37],RBF神经网络译码器在传输信道的输出端工作在$w\times n_0$(w 为窗口的大小)位的滑动窗上。例如当 w 为偶数，在无干扰传输时，RBF 神经网络译码器的输入端为$w\times n_0$位的加窗比特序列R_i，R_i等同于卷积码编码器输出的相应的$w\times n_0$比特序列C_i。C_i包括有关码字B_i的w个加窗码字，$R_i=C_i=[B_{i-w/2}\cdots B_i\cdots B_{i+w/2-1}]$。在时序的控制下，RBF 神经网络译码器处理输入码字R_i，先得到与码字B_i相关联的信息比特b_i，然后移动窗口一个位置，新加入一个码字$B_{i+w/2}$，丢弃码字$B_{i-w/2}$，那么就可以处理下一个码字C_{i+1}。按照这种方法，半无限长的卷积码就可以看成一系列互相重叠连接起来的分组码，如图 3-3(b)。



(a)　RBF 神经网络纠错系统



(b) 接收到的码字结构

图 3-3　滑动窗口

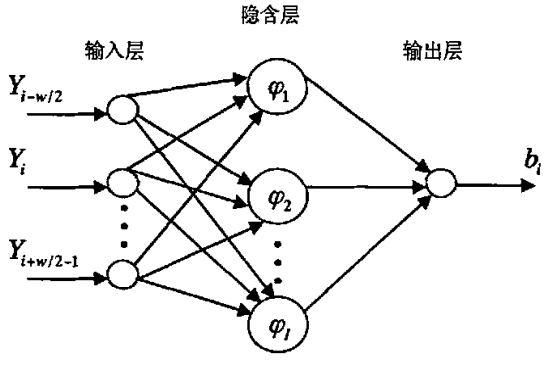


图 3-4　RBF 神经网络译码器结构

图 3-4 中 RBF 神经网络译码器输入层为接收序列经过加窗后得到的序列 R_i。隐含层中的拟和中心为多步状态转移的输出部分。多步状态转移的输出部分确定方法如下：假设 R_i 中包含 u 个信息，那么这 u 个信息可组成 2^u 种信息序列，将这 2^u 种信息序列送入卷积码编码器，根据不同的初始状态（假设状态数目为 s），可得到 $s \times 2^u$ 种状态转移输出，这些输出即为多步状态转移的输出。由于 RBF 网

络译码器是对接收序列中的子码逐个译码，所以输出层单元为 1 个。

经过网络训练后，研究 RBF 神经网络权值发现，随着 SNR 的提高，网络权值越来越趋近于某些固定的值，这些固定的值和拟和中心各比特位上的值有着密切关系。直接用 RBF 的各个拟和中心的相同位置输入信息的值组成 RBF 网络权值矩阵，即将 W 的值固定作为网络的权值，可以得到较好的译码性能。这样可以使运算方法简洁，硬件实现更容易，现实可操作性更强。

3.2.2 仿真结果与分析

仿真采用(2，1，2)系统卷积码，如图 3-5 所示；采用 BPSK，AWGN 信道模型；图 3-4 所示的译码方案，采用固定权值的 RBF 神经网络译码模型，分别对窗口大小 w=5、6、7 时，RBF 神经网络译码器对码序列中不同位置的子码译码时所表现出来的性能进行了仿真。译码器的构成为：输入节点为 $2\times w$ 个，隐层单元为 4×2^w 个，输出单元为 1 个($k_0=1$)。加窗后的码序列中的信息位个数即为窗口大小 w，(2，1，2)系统卷积码的 n_0 为 2，所以输入节点为 $2\times w$ 个；又因为(2，1，2)系统卷积码有 4 个状态，信息位可能的组合有 2^w 个，由这 2^w 种的信息序列经过(2，1，2)系统卷积码编码器可得到 4×2^w 个可能的码序列，所以隐含层单元为 4×2^w 个；输出层单元为 1 个。

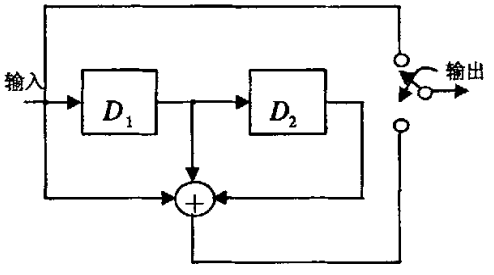


图 3-5　(2，1，2)系统卷积码编码器

(1) 窗口大小为 5 的仿真实验

由图 3-6 看出，误码率为 10^{-4} 时，当译第三位的时候，可以获得 1dB 的编码增益。随着所译位置越来越靠近码字的中间位置，译码性能得到很好的改善，

纠错范围逐渐扩大。对于译第三位，当信噪比 SNR>3.5dB 时，RBF 神经网络译码器具有纠错能力。

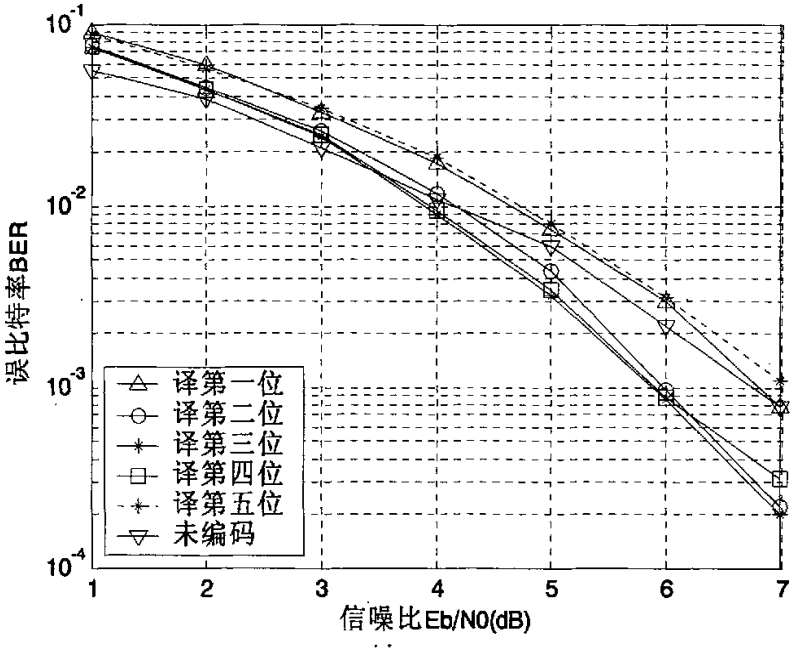


图 3-6 窗口大小为 5 时的 RBF 神经网络译码性能

(2) 窗口大小为 6 的仿真实验

由图 3-7 看出，误码率为10^{-4}时，当译第四位的时候，可以获得 1.4dB 的编码增益。随着所译位置越来越靠近码字的中间位置，译码性能得到很好的改善，纠错范围逐渐扩大。对于译第四位，当信噪比 SNR>1.5dB 时，RBF 神经网络译码器具有纠错能力。

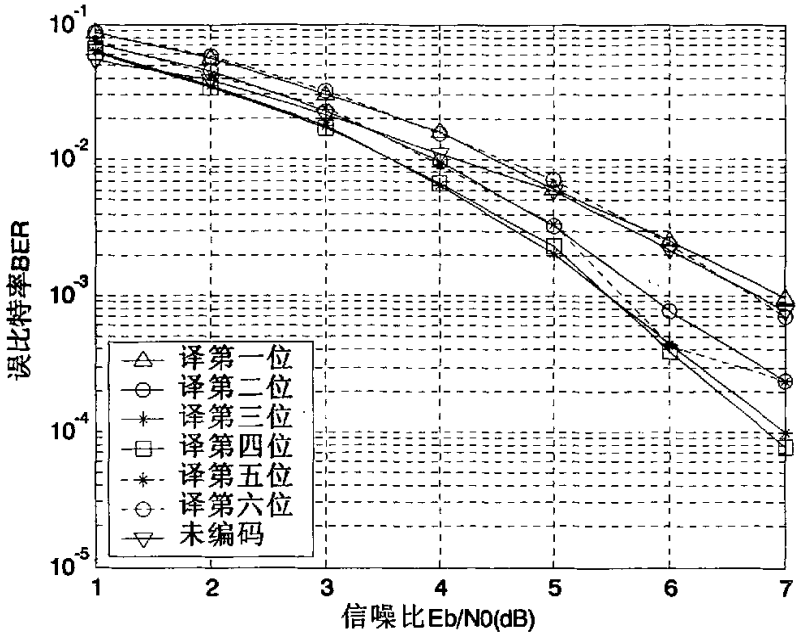


图 3-7 窗口大小为 6 时的 RBF 神经网络译码性能

(3) 窗口大小为 7 的仿真实验

由图 3-8 看出，误码率为10^{-4}时，当译第四位的时候，可以获得 1.8dB 的编码增益。随着所译位置越来越靠近码字的中间位置，译码性能得到很好的改善，纠错范围逐渐扩大。对于译第四位，当信噪比 SNR>1dB 时，RBF 神经网络译码器具有纠错能力。

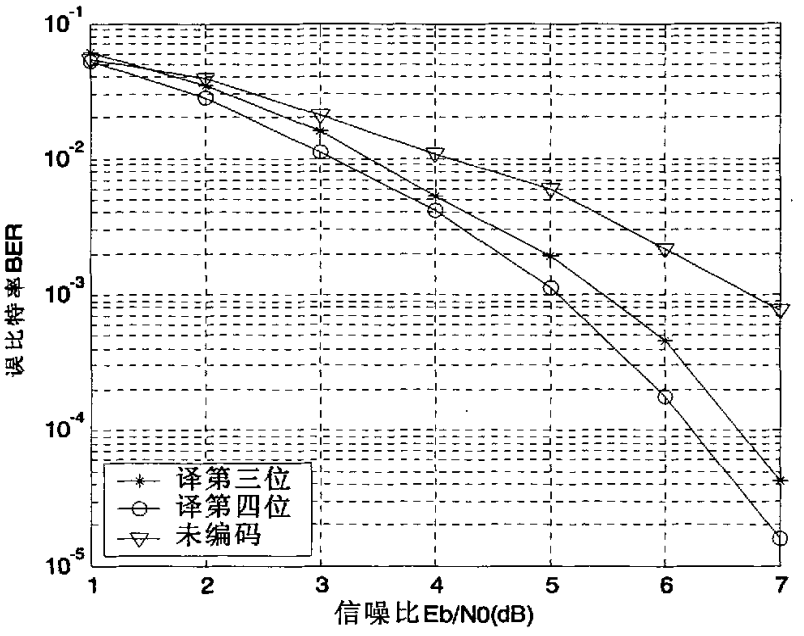


图 3-8 窗口大小为 7 时的 RBF 神经网络译码性能

图 3-6~8 显示出 $w=5$、6、7 时 RBF 神经网络译码器对不同位置的子码译码时的译码性能。结果表明，译处于中间位置上的子码效果最好。这是因为卷积码编码时，本组的 n_0-k_0 个校验元不仅与本组的 k_0 个信息元有关，而且还与以前各时刻输入至编码器的信息组有关。同样在卷积码译码时，不仅要根据此时刻输入到译码器的子码，而且还要根据以后很长一段时间内，收到的各子码，才能译出一个子码信息元。在译处于较前和较后位置的子码时，更大的降低了这些子码与前后一段时间内各子码间的相关性，因此译处于较前和较后位置上的子码效果没有译处于中间位置上的子码效果好，所译位置越靠中间，效果越好。

(4) 不同窗口大小的 RBF 神经网络译码性能比较

由图 3-9 可以看出，随着 w 的增大，译码的误比特率不断降低；同时，可以纠错的范围越来越大。

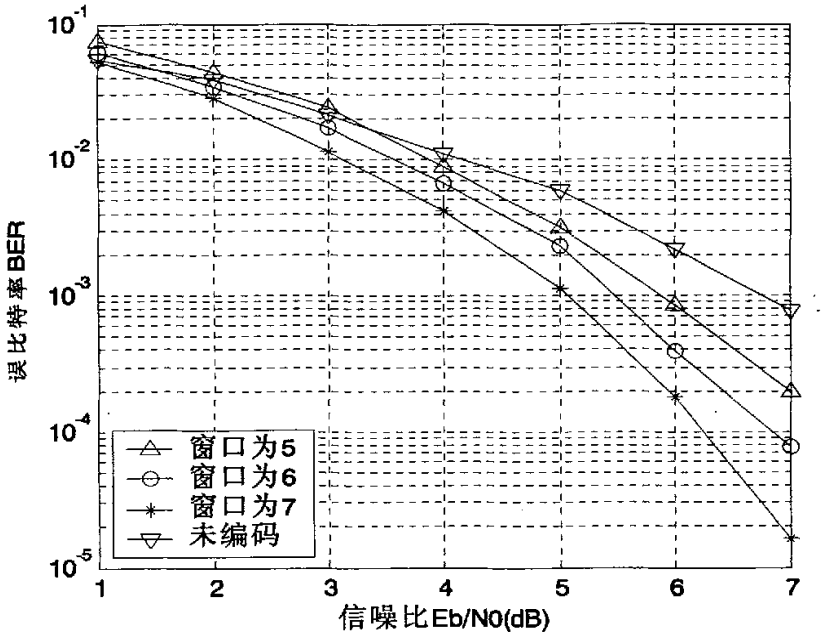


图 3-9 不同窗口大小的 RBF 神经网络译码性能比较

这是因为随着窗口的增大，提高了 R_i 中子码间的相关性，使它能更多的继承卷积码子码间的相关性，同时更加的接近半无限长序列，译码的可靠性增加。但是，由于 RBF 隐层单元数目随着窗口中的信息比特数目的增加而增多，运算复杂度将会提高，因此所选择的窗口不能过大。在仿真的时候，也对(2，1，2)非系统卷积码进行了类似的仿真，也得到了相同的结论。

图 3-10 中给出了不同窗口大小的 RBF 神经网络译码性能、Viterbi 硬判决和 Viterbi 软判决的性能[1]，并将它们进行了比较。从图中可以看出，对于所有实验的窗口大小，RBF 神经网络译码器的性能都不如 Viterbi 软判决译码器的性能；但与 Viterbi 硬判决译码器的性能相比，RBF 神经网络译码器的性能有一定的改善，性能要好于 Viterbi 硬判决译码器的性能。随着 SNR 的提高和译码窗口的增大，RBF 神经网络译码器的性能越来越好于 Viterbi 硬判决译码器的性能，并且越来越接近 Viterbi 软判决译码器的性能。其原因是，RBF 神经网络译码器是局部匹配的，而 Viterbi 算法是一种迭代算法，它对所有状态路径的匹配是全局性的。Viterbi 译码是具有记忆存储的，它存储了每一状态路径及其似然值作为判断

准则，而 RBF 神经网络译码器是无记忆的，它只考虑窗口范围内的值，并对其进行译码。

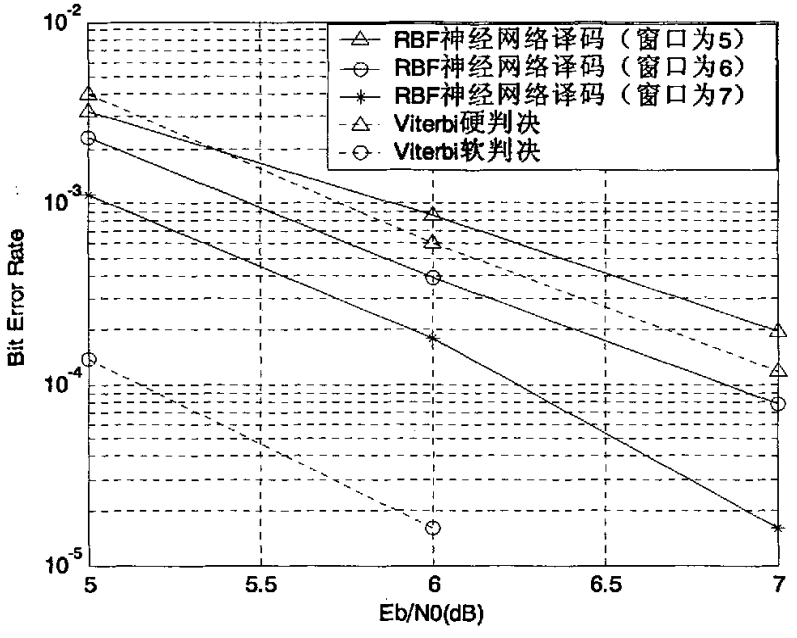


图 3-10 RBF 神经网络译码算法和 Viterbi 算法的译码性能比较

3.3 卷积 Turbo 码的 RBF 神经网络译码算法

3.3.1 无外信息的译码算法

由 3.2 节的实验结果可知，卷积码的 RBF 神经网络译码算法具有并行处理能力和良好的译码性能。将此训练好的 RBF 网络译码器直接嵌入 Turbo 码译码器，替代图 2-3 中两个分量译码器，选择(n_0, k_0, m)系统卷积码编码器来代替图 2-2 中的两个分量编码器。在不提取外信息的时候，将分量译码器 1 的译码输出通过交织，直接送入分量译码器 2，并与相应的校验位组成分量译码器 2 的输入，分量译码器 2 的译码输出通过解交织，直接送入分量译码器 1，经过多次迭代，译码性能不再提高。

仿真用的卷积 Turbo 码由两个(2，1，2)系统卷积码编码器构成，采用伪随机交织器；用 BPSK，AWGN 信道模型；图 3-4 所示的译码方案，采用固定权值

的 RBF 神经网络译码模型，分别对窗口大小 w =5、6、7 时，RBF 神经网络译码器对码序列中处于中间位置的子码译码时所表现出来的性能进行了仿真。

图 3-11 中窗口大小为 5，由图可知，没有外信息的神经网络算法具有比较高的编码增益。在误码率为10^{-4}时，5 次迭代译码比 1 次迭代译码可以获得 1.5dB 的编码增益。经过 5 次完全迭代译码后，译码性能趋于稳定，随着迭代次数进一步的增加，性能改善较小。

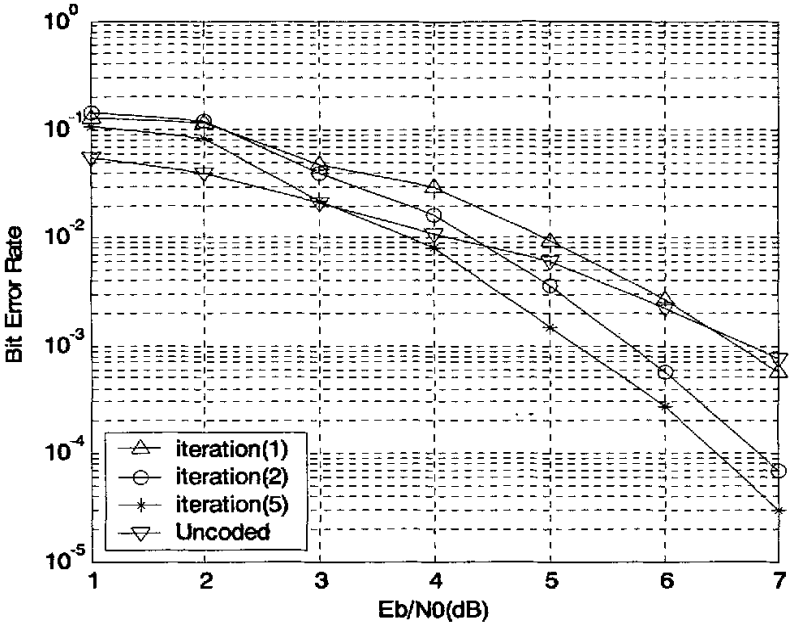


图 3-11 窗口大小为 5 时的 RBF 神经网络译码性能

通过实验仿真，发现随着译码窗口的增大，分量译码器 1 的输出中有很多值趋近 0，不能作为下一分量译码器 2 的输入，在 w =6、7 时，无法正常译码。同时，由于采用的是滑动窗口方法，在窗口较小的时候，势必会更多的降低卷积码子码之间的相关性，从而影响卷积 Turbo 码的子码的相关性，造成译码不可靠，译码准确度低。

3.3.2 修改后无外信息的译码算法

由 3.3.1 节的实验结果可知，卷积 Turbo 码的 RBF 神经网络译码算法具有

并行处理能力和良好的译码性能。但随着译码窗口的增大，分量译码器 1 的输出中有大量的值趋近 0，当这些值作为下一分量译码器 2 的输入，会严重影响卷积 Turbo 码的译码，使其误比特率增高。所以在窗口增大的时候，用卷积码 RBF 神经网络译码器直接嵌入卷积 Turbo 码译码器的方法是不合适的。因此，需对上一节所述的神经网络模型进行调整，以使算法得以更好的实现。

3.3.2.1 无外信息的 RBF 神经网络译码器

卷积 Turbo 码的编码输出经过 BPSK 调制后所得到的码序列 $C'=(c'_1,c'_2,\cdots,c'_i,\cdots,c'_n)$，其中 $c'_i=(u'_i\,p'^1_i\,p'^2_i)$，$u'$、$p'^1$ 和 p'^2 分别为信息位，分量编码器 1 产生的校验位和分量编码器 2 产生的校验位。对应于分量编码器 1 的码序列为 $C'_1=(c'_{11},c'_{21},\cdots,c'_{i1},\cdots,c'_{n1})$，其中 $c'_{i1}=(u'_i\,p'^1_i)$。编码后的输出序列经过 BPSK 调制和均值为 0，方差为 σ^2 的 AWGN 信道传输。对应的接收序列为 $R=(r_1,r_2,\cdots,r_i,\cdots,r_n)$，该序列对应于分量译码器 1 的接收序列为 $R^1=(r_{11},r_{21},\cdots,r_{i1},\cdots,r_{n1})$。假定窗口大小 w 为偶数，经过 BPSK 调制后的分量码编码器编码输出的所有许用码为 $C=(c_1,c_2,\cdots,c_j,\cdots,c_{2^{w\times n_0}})$，其中 $c_j=(u_{j1}p_{j1},u_{j2}p_{j2},\cdots,u_{jk}p_{jk},\cdots,u_{jw}p_{jw})$，$u$ 和 p 分别代表信息位和校验位；R^1 用 3.2.1 节中的方法经加窗后可得到码字集合$\{R_i\}$。由于传输信道为 AWGN 信道，R_i 是 c_j 经过噪声信道加噪而来的概率密度函数为：

$$P(R_i\,|\,c_j)=\left(\frac{1}{\sigma\sqrt{2\pi}}\right)^{w\times n_0}\exp\left(-\frac{\|R_i-c_j\|^2}{2\sigma^2}\right),\tag{3-4}$$

所以，对于信息位 $R_i^{u_f}$ $(f=w/2)$，加噪后的信息位是由 $u_f=+1$ 和 $u_f=-1$ 转化而来的概率计算如下：

$$P(u_f=+1\,|\,R_i)=\sum_{c_j\in S_f^{+1}}P(R_i\,|\,c_j),\tag{3-5}$$

$$P(u_f=-1\,|\,R_i)=\sum_{c_j\in S_f^{-1}}P(R_i\,|\,c_j),\tag{3-6}$$

式中 S_f^{+1}，S_f^{-1} 分别是码集$\{C\}$中满足条件 $u_f=+1$，$u_f=-1$ 的集合。

利用 3.3.1 节中所采用的 RBF 神经网络方法，可以看出上对于卷积码的 RBF 神经网络译码，其译码器的输出利用式(3-1)、(3-2)、(3-3)、(3-5)和(3-6)可以得到输出

$$y_f = \sum_{j=1}^{I} W_{jf}\varphi\left(X, T_j\right)$$

$$= \frac{P(u_f = +1|R_i) - P(u_f = -1|R_i)}{\left(\frac{1}{\sigma\sqrt{2\pi}}\right)^{w \times n_0}}, \tag{3-7}$$

由(3-7)式，可以得出，正是因为分子为 0，也就是说加噪后的信息位是由 $u_f = +1$和$u_f = -1$转化而来的概率几乎相等，造成了 3.3.1 节中的 RBF 神经网络译码器不能直接嵌入卷积 Turbo 码的译码器。为了避免这种情况的发生，将 RBF 神经网络译码器的输出进行如下改进：

译码器的接收量只可能由+1 或-1 经加噪后得到，由贝叶斯公式可得

$$P(u_f = +1) = \frac{P(u_f = +1|R_i)}{P(u_f = +1|R_i) + P(u_f = -1|R_i)}, \tag{3-8}$$

$$P(u_f = -1) = \frac{P(u_f = -1|R_i)}{P(u_f = +1|R_i) + P(u_f = -1|R_i)}。 \tag{3-9}$$

当 $P(u_f = +1) \geq P(u_f = -1)$，RBF 神经网络译码器的输出为 $P(u_f = +1)$；

当 $P(u_f = +1) < P(u_f = -1)$，RBF 神经网络译码器的输出为 $-P(u_f = -1)$。

继续推导，由于 RBF 神经网络的输出 $Y=GW$，权值 W 是由许用码集 $\{C\}$ 相同位置上相关的信息位的值构成，是固定的，其中的元素均为+1、-1； 取高斯函数 G 为“基函数”。根据式(3-1)、(3-5)和式(3-6),我们可以得到：

连接权值为+1 的输出层单元的输出为

$$y_{f_u}^{+1} = \sum_{c_j \in S_f^{+1}} G(\| R_i - c_j \|), \tag{3-10}$$

连接权值为-1 的输出层单元的输出为

$$y_{f_u}^{-1} = \sum_{c_j \in S_f^{-1}} G(\| R_i - c_j \|), \tag{3-11}$$

根据式(3-10)和式(3-11)，RBF 神经网络译码器的输出可以表示为

$$y_{f_u} = \begin{cases} \dfrac{y_{f_u}^{+1}}{y_{f_u}^{+1} + y_{f_u}^{-1}} & when \dfrac{y_{f_u}^{+1}}{y_{f_u}^{+1} + y_{f_u}^{-1}} \geq \dfrac{y_{f_u}^{-1}}{y_{f_u}^{+1} + y_{f_u}^{-1}} \\ -\dfrac{y_{f_u}^{-1}}{y_{f_u}^{+1} + y_{f_u}^{-1}} & when \dfrac{y_{f_u}^{+1}}{y_{f_u}^{+1} + y_{f_u}^{-1}} < \dfrac{y_{f_u}^{-1}}{y_{f_u}^{+1} + y_{f_u}^{-1}} \end{cases}, \tag{3-12}$$

同理也可以推导出校验位计算式 y_{f_p}。

由上面的分析和推导，可以得到改进后的 RBF 神经网络译码器，如图 3-12 所示，图中 b_i 为译出的信息位，p_i 为译出的校验位。

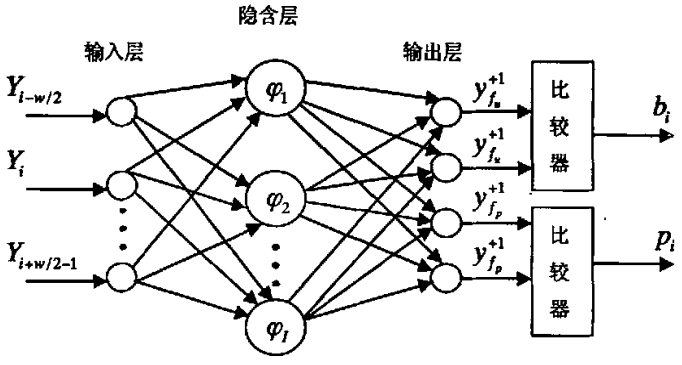


图 3-12 无外信息的 RBF 神经网络译码器模型

3.3.2.2 仿真结果与分析

仿真用的卷积 Turbo 码由两个(2，1，2)系统卷积码编码器构成，采用伪随机交织器；用 BPSK，AWGN 信道模型；图 3-12 所示的译码方案，采用固定权值的 RBF 神经网络译码模型，分别对窗口大小 w =5、6、7 时，RBF 神经网络译码器对码序列中处于中间位置的子码译码时所表现出来的性能进行了仿真。

(1) 窗口大小为 5 的仿真实验

图 3-13 中窗口大小为 5，由图可知，没有外信息的神经网络算法具有比较高的编码增益。在误码率为 10^{-3} 时，1 次迭代译码可以获得 0.4dB 的编码增益，5 次迭代可以获得 1.5dB 的编码增益。经过 5 次完全迭代译码后，译码性能趋于稳定，随着迭代次数进一步的增加，性能改善较小。

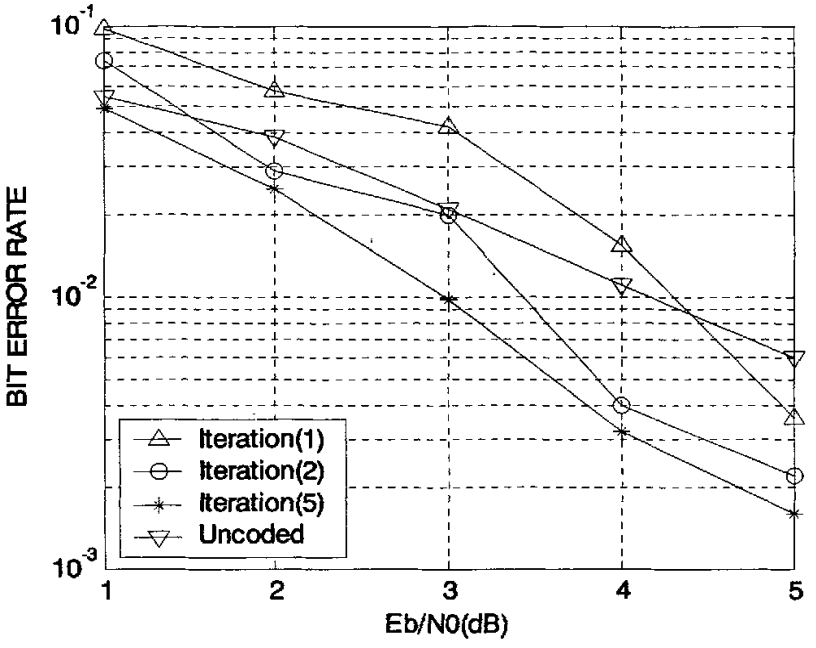


图 3-13 窗口大小为 5 时的 RBF 神经网络译码性能

(2) 窗口大小为 6 的仿真实验

图 3-14 中窗口大小为 6，由图可知，没有外信息的神经网络算法具有比较高的编码增益。在误码率为10^{-3}时，1 次迭代译码可以获得 0.8dB 的编码增益，5 次迭代可以获得 1.8dB 的编码增益。经过 5 次完全迭代译码后，译码性能趋于稳定，随着迭代次数进一步的增加，性能改善较小。

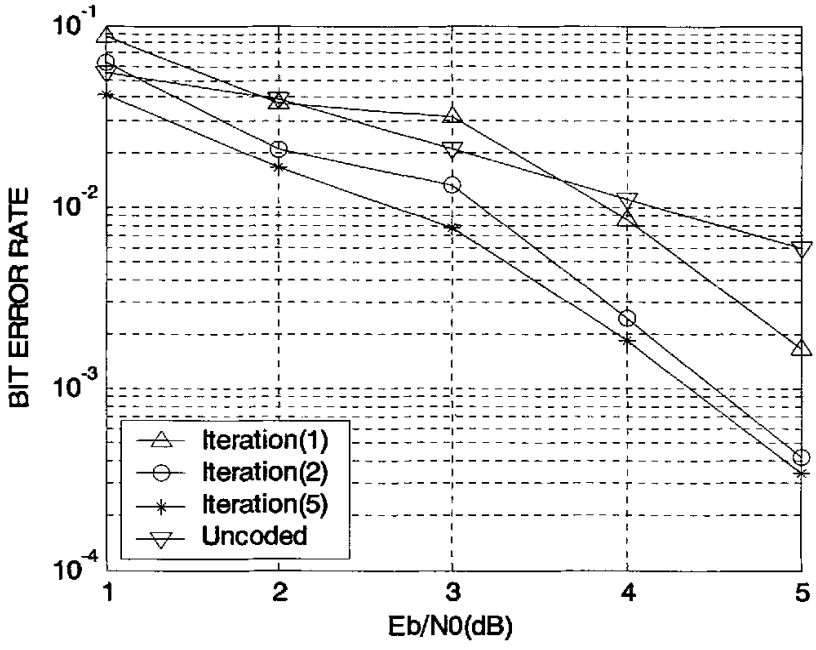


图 3-14 窗口大小为 6 时的 RBF 神经网络译码性能

(3) 窗口大小为 7 的仿真实验

图 3-15 中窗口大小为 7，由图可知，没有外信息的神经网络算法具有比较高的编码增益。在误码率为10^{-3}时，1 次迭代译码可以获得 1dB 的编码增益，5 次迭代可以获得 2.4dB 的编码增益。经过 5 次完全迭代译码后，译码性能趋于稳定，随着迭代次数进一步的增加，性能改善较小。

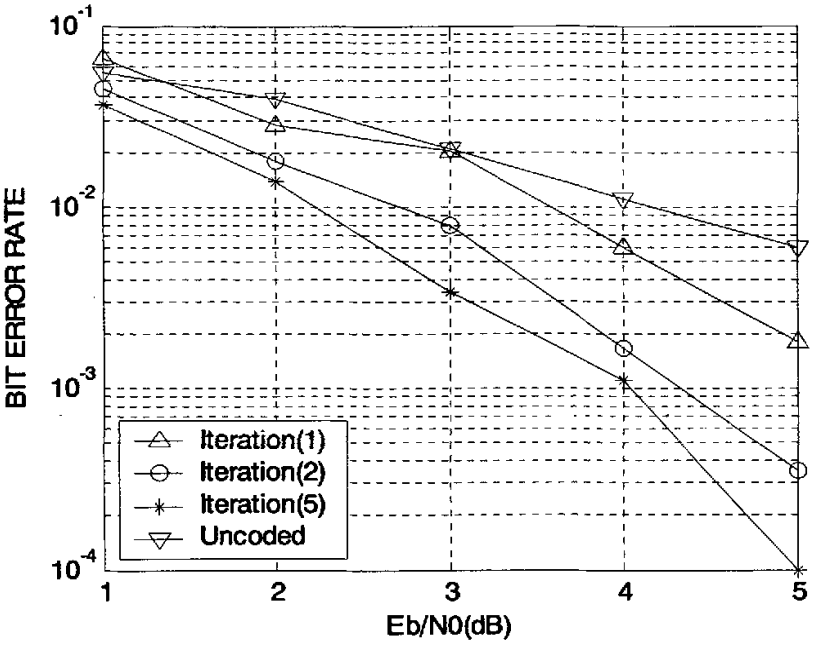


图 3-15 窗口大小为 7 时的 RBF 神经网络译码性能

(4) 不同窗口大小的 RBF 神经网络译码性能比较

由图 3-16 可以看出，随着 w 的增大，译码的误比特率不断降低。这是因为随着窗口的增大，提高了 R_i 中子码间的相关性，使它能更多的继承卷积 Turbo 码子码间的相关性，同时更加的接近半无限长序列，译码的可靠性增加。但是，由于 RBF 隐层单元数目随着窗口中的信息比特数目的增加而增多，运算复杂度将会提高，因此所选择的窗口不能过大。

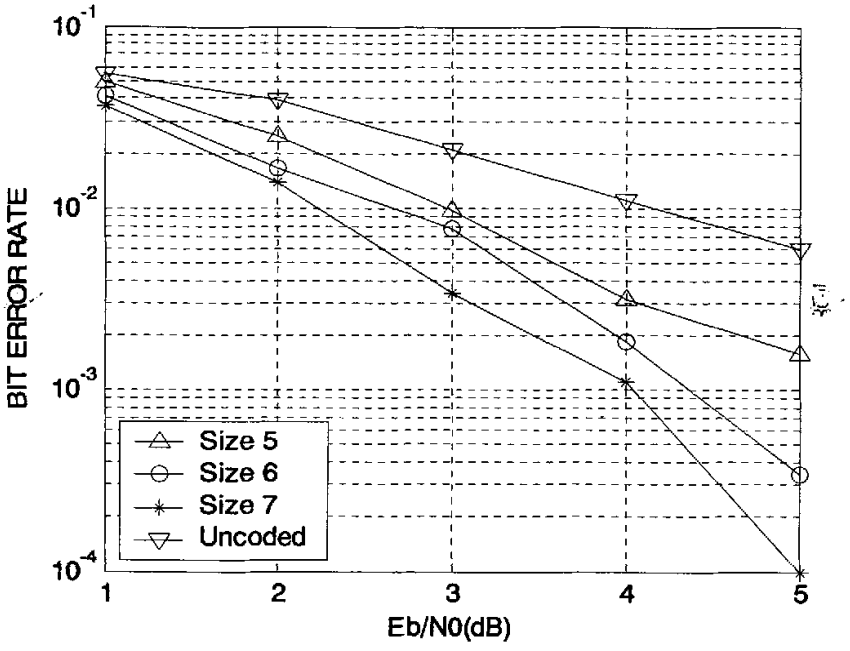


图 3-16 不同窗口大小的 RBF 神经网络译码性能比较

3.3.3 有外信息的译码算法

在 3.3.2 节的基础上，对卷积 Turbo 码的神经网络译码器做进一步的研究。由于传统的 Turbo 码译码器是利用外信息来进行迭代译码算法。3.3.2 节所采用的无外信息 RBF 神经网络译码算法，势必会丢失一部分信道信息，不能充分利用信道信息，给译码结果的准确度带来一定的影响。基于上述考虑，提出了带有外信息的卷积 Turbo 码的 RBF 神经网络译码器。

3.3.3.1 有外信息的 RBF 神经网络译码器

算法中外信息的推导过程如下：

卷积 Turbo 码的编码输出经过 BPSK 调制后所得到的码序列 $C'=(c'_1,c'_2,\cdots,c'_i,\cdots,c'_n)$，其中 $c'_i=(u'_i p'^1_i p'^2_i)$，$u'$、$p'^1$ 和 p'^2 分别为信息位，分量编码器 1 产生的校验位和分量编码器 2 产生的校验位。对应于分量编码器 1 的码序列为 $C'_1=(c'_{11},c'_{21},\cdots,c'_{i1},\cdots,c'_{n1})$，其中 $c'_{i1}=(u'_i p'^1_i)$。编码后的输出序列经过 BPSK 调制和均值为 0，方差为 σ^2 的 AWGN 信道传输。对应的接收序列为

$R=(r_1,r_2,\cdots,r_i,\cdots,r_n)$，该序列对应于分量译码器 1 的接收序列为 $R^1=(r_{11},r_{21},\cdots,r_{i1},\cdots,r_{n1})$。假定窗口大小 w 为偶数，经过 BPSK 调制后的分量码编码器编码输出的所有许用码为 $C=(c_1,c_2,\cdots,c_j,\cdots,c_{2^{k\times w}})$，其中 $c_j=(u_{j1}p_{j1},u_{j2}p_{j2},\cdots,u_{jk}p_{jk},\cdots,u_{jw}p_{jw})$，$u$ 和 p 分别代表信息位和校验位；R^1 用 3.2.1 节中的方法经加窗后可得到码字集合$\{R_i\}$。该码字集合经过译码器译码后的软输出为 $D=(d_{1h},d_{2h},\cdots,d_{ih},\cdots d_{nh})$，其中 $h=w/2$。d_{ih} 的定义[1][5]如下：

$$d_{ih}=\ln\frac{P(u_h=+1|R_i)}{P(u_h=-1|R_i)}, \tag{3-13}$$

其中，u_h 为信息位

$$P(u_h=+1|R_i)=\sum_{c_j\in S_h^{+1}}P(R_i|c_j), \tag{3-14}$$

$$P(u_h=-1|R_i)=\sum_{c_j\in S_h^{-1}}P(R_i|c_j), \tag{3-15}$$

式(3-14)和(3-15)中 S_h^{+1}，S_h^{-1} 分别是码集$\{C\}$中满足条件 $u_h=+1$，$u_h=-1$ 的集合。

根据贝叶斯公式，在假定码字等概传输的前提下，式(3-13)可以表示为：

$$d_{ih}=\ln\frac{\sum_{c_j\in S_h^{+1}}P(R_i|c_j)}{\sum_{c_j\in S_h^{-1}}P(R_i|c_j)}, \tag{3-16}$$

式中

$$P(R_i|c_j)=\left(\frac{1}{\sigma\sqrt{2\pi}}\right)^{w\times n_0}\exp\left(-\frac{\|R_i-c_j\|^2}{2\sigma^2}\right), \tag{3-17}$$

对式(3-16)进一步推导，可得：

$$d_{ih}=\ln\frac{\sum_{c_j\in S_h^{+1}}P(R_i|c_j)}{\sum_{c_j\in S_h^{-1}}P(R_i|c_j)}$$

$$=\ln\frac{\sum_{c_j\in S_h^{+1}}\left(\frac{1}{\sigma\sqrt{2\pi}}\right)^{w\times n_0}\exp\left(-\frac{\|R_i-c_j\|^2}{2\sigma^2}\right)}{\sum_{c_j\in S_h^{-1}}\left(\frac{1}{\sigma\sqrt{2\pi}}\right)^{w\times n_0}\exp\left(-\frac{\|R_i-c_j\|^2}{2\sigma^2}\right)}$$

$$= \ln \frac{\sum_{c_j \in CS_h^{+1}} \exp(-\frac{\| R_i - c_j \|^2}{2\sigma^2})}{\sum_{c_j \in CS_h^{-1}} \exp(-\frac{\| R_i - c_j \|^2}{2\sigma^2})}, \tag{3-18}$$

对式(3-18)进一步推导，可得：

$$d_{ih} = \frac{2}{\sigma^2}(r_{ih} + l_{ih}), \tag{3-19}$$

式中 l_{ih} 为外信息。

然后按照下列规则进行判决：

$$y_{ih} = \begin{cases} 1 & d_{ih} \geq 0 \\ 0 & d_{ih} < 0 \end{cases}, \tag{3-20}$$

继续推导，由于 RBF 神经网络的输出 $Y=GW$，权值 W 是由许用码集 $\{C\}$ 相同位置上相关的信息位的值构成，是固定的，其中的元素均为+1、-1； 取高斯函数 G 为“基函数”。根据式(3-1)、(3-14)和式(3-15),我们可以得到：

连接权值为+1 的输出层单元的输出为：

$$y_{ih_u}^{+1} = \sum_{c_j \in CS_h^{+1}} G(\| R_i - c_j \|), \tag{3-21}$$

连接权值为-1 的输出层单元的输出为：

$$y_{ih_u}^{-1} = \sum_{c_j \in CS_h^{-1}} G(\| R_i - c_j \|), \tag{3-22}$$

将式(3-21)和式(3-22)代入(3-18)，可得：

$$d_{ih} = \ln \frac{y_{ih_u}^{+1}}{y_{ih_u}^{-1}}, \tag{3-23}$$

所以，利用式(3-19)和式(3-23)，我们可得

$$l_{ih} = \frac{\sigma^2}{2} \ln \left(\frac{y_{ih_u}^{+1}}{y_{ih_u}^{-1}} \right) - r_{ih_u} \tag{3-24}$$

同理也可以推导出校验位计算式 y_{ih_p}。

由上面的分析和推导，可以得到改进后的带有外信息的 RBF 神经网络译码器，只是将图 3-12 中 RBF 神经网络译码器的输出端进行调整，加入了除法器，如图 3-17 所示，图中 b_i 为译出的信息位，p_i 为译出的校验位。

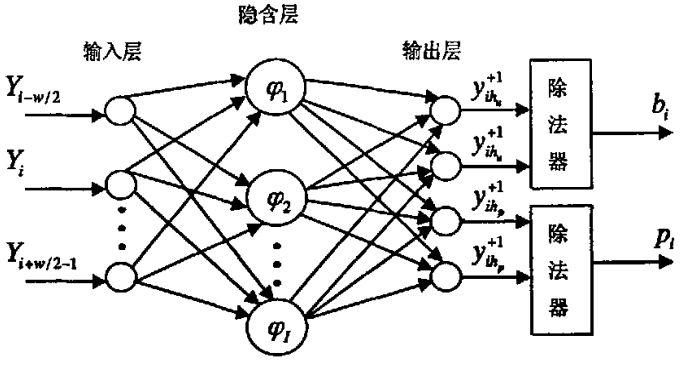


图 3-17 带外信息的 RBF 神经网络译码器模型

3.3.3.2 仿真结果与分析

仿真用的卷积 Turbo 码由两个(2，1，2)系统卷积码编码器构成，采用伪随机交织器；用 BPSK，AWGN 信道模型；图 3-17 所示的译码方案，采用固定权值的 RBF 神经网络译码模型，对窗口大小 $w=7$ 时，RBF 神经网络译码器对码序列中处于中间位置的子码译码时所表现出来的性能进行仿真。

图 3-18 中窗口大小为 7，由图可知，有外信息的神经网络算法具有比较高的编码增益。在误码率为10^{-3}时，4 次迭代译码可以获得 1.8dB 的编码增益，10 迭代可以获得 2.8dB 的编码增益。经过 10 次完全迭代译码后，译码性能趋于稳定，随着迭代次数的增加，性能改善较小。

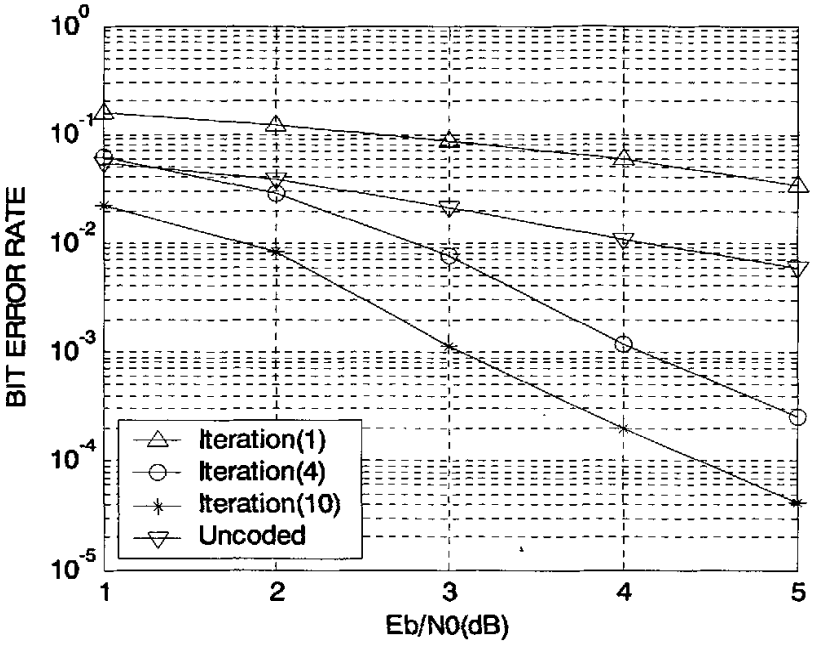


图 3-18 带有外信息的 RBF 神经网络译码性能

3.4 不同 RBF 神经网络译码器的性能比较

图 3-19 为窗口大小为 7 时，不同 RBF 神经网络译码器的性能比较图。从图中可以看出，卷积 Turbo 码的 RBF 神经网络译码算法与卷积码的 RBF 神经网络译码算法相比，译码性能有很大的提高。在误码率为10^{-3}时，卷积 Turbo 码无外信息的 RBF 神经网络译码算法比卷积码的 RBF 神经网络译码算法要多获得 1dB 的编码增益；卷积 Turbo 码有外信息的 RBF 神经网络译码算法比卷积码的 RBF 神经网络译码算法要多获得 2dB 的编码增益。在误码率为10^{-4}时，有外信息的 RBF 神经网络译码算法比无外信息的 RBF 神经网络译码算法要多获得 1dB 的编码增益。这是由于外信息的提取，可以更好的纠正噪声信道的对译码的干扰，通过对外信息不断的改善，来提高译码的可靠性，达到提高译码性能的目的。

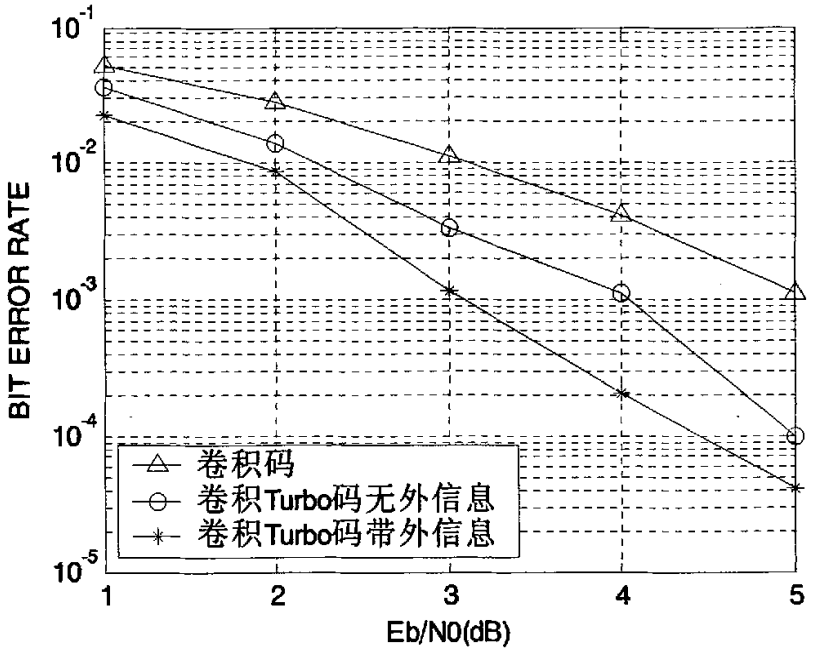


图 3-19 不同 RBF 神经网络译码器的性能

对于分量编码器由(n_0，k_0，m)组成的卷积 Turbo 码(m=v)，在译码窗口大小为 w 的时候，无外信息的情况下，每译出一位信息要做$(w+1)\times 2^v$ 次加法和 2^v+2 次乘(除)法；有外信息的情况下，每译出一位信息要做次$(w+1)\times 2^v+1$加法和2^v+1次乘(除)法。

表 3.1 给出了不同算法之间的复杂度的比较[35]，其中 v 为编码寄存器的数目，w 为滑动窗口的大小。

表 3.1　不同译码算法的复杂度比较（单位：次数）

操作	求最大值	查找	加法	乘（除）法
MAP	0	0	4×2^{v}	$6\times 2^{v}+1$
Max-Log-MAP	$5\times 2^{v}-2$	0	$10\times 2^{v}+11$	8
RBF 神经网络（无外信息）	0	0	$(w+1)\times 2^{v}$	$2^{v}+2$
RBF 神经网络（有外信息）	0	0	$(w+1)\times 2^{v}+1$	$2^{v}+1$

对于由编码寄存器为 2 的卷积码所组成的卷积 Turbo 码，利用 RBF 神经网络进行译码时，在运算复杂度方面，无外信息方法和有外信息方法的运算次数非常接近。但和 MAP、Max-Log-MAP 算法相比时，可以明显的看出，运算的复杂度有所降低。根据对做相同次数加法和乘法的时间统计，做相同次数乘法所用的时间为做相同次数加法所用时间的 5 倍，这就说明做相同次数的乘法运算的复杂度是做相同次数加法运算复杂度的 5 倍。同时，求最大值运算的复杂性等同于加法运算的复杂性。用表中的公式计算，在窗口大小 $w=5$，RBF 神经网络译码算法的运算复杂度分别是 MAP 算法的 5/14、Max-Log-MAP 算法的 5/11；在窗口大小 $w=6$，RBF 神经网络译码算法的运算复杂度分别是 MAP 算法的 5/13、Max-Log-MAP 算法的 1/2；在窗口大小 $w=7$，RBF 神经网络译码算法的运算复杂度分别是 MAP 算法的 5/12、Max-Log-MAP 算法的 5/9。随着窗口的增大，RBF 神经网络译码算法的运算复杂度将会提高，同时，误比特率将会降低，译码性能得到很好的改善。

第 4 章　总结

Turbo 译码算法是 Turbo 码译码系统的关键部分，Turbo 码的性能主要取决于 Turbo 码的译码算法。Turbo 码的译码算法主要有两大类。一类是基于最大后验概率(MAP)的软输入算法；另一类 Turbo 码算法是基于 Viterbi 算法(B-SOVA)和 List－SOVA 等。

本文从考虑降低卷积 Turbo 码的译码复杂度，便于硬件实现的角度出发，提出卷积 Turbo 码的 RBF 神经网络译码算法。通过利用 RBF 神经网络强大的并行处理能力和非线性分类能力，降低卷积 Turbo 码的译码复杂度。在利用 RBF 神经网络进行译码的时候，讨论了三种译码算法。分别是：

(1) 卷积码的 RBF 神经网络译码算法。采用该算法的时候，引入了滑动窗口的方法，将卷积码译码和 RBF 神经网络结合，用并行处理的方式降低了译码时延。经过仿真可知，随着窗口的增大，这种译码算法的性能会越来越接近 Viterbi 软判决译码的性能。

(2) 没有外信息的卷积 Turbo 码的 RBF 神经网络译码算法。在卷积码的 RBF 神经网络译码算法的基础上，将这种译码算法推广到卷积 Turbo 码译码。将卷积码的 RBF 神经网络译码器直接嵌入卷积 Turbo 码的译码系统时，在窗口较小的时候，可以实现卷积 Turbo 码的译码；在窗口增大的时候，就无法实现卷积 Turbo 码的译码。鉴于这个问题，提出了修改后的没有外信息的卷积 Turbo 码的 RBF 神经网络译码算法。这种改善，使得在滑动窗口增大时，仍能对卷积 Turbo 码进行译码，并获得更好的译码性能。同时，译码复杂度相比 MAP 算法和 Max-Log-MAP 算法都有很大程度的降低。

(3) 带有外信息的卷积 Turbo 码的 RBF 神经网络译码算法。在没有外信息的卷积 Turbo 码的 RBF 神经网络译码算法的基础上，进行了更深一步的研究，

提出带有外信息的方法。利用这种方法，在不提高复杂度的情况下，可以获得更高的编码增益，获得更好的译码性能。

卷积 Turbo 码的 RBF 神经网络译码算法是一种新的译码算法。该算法具有并行处理能力，能够对卷积 Turbo 码进行有效的迭代译码，译码性能较好，运算复杂度相比 MAP 算法和 Max-Log-MAP 算法都有很大程度的降低。因此，该方法使得卷积 Turbo 码的译码算法更为简洁，易于硬件实现和实际系统中的应用。

提出的卷积 Turbo 码的 RBF 神经网络译码算法目前只考虑了由分量编码寄存器为 2 的卷积码所组成的卷积 Turbo 码的情况。在寄存器数目增大的情况下，如何更好的将上述算法应用到译码中还有待探讨。

在现有的基础上，后续的发展方向有：

(1) 考虑编码寄存器数目较大的卷积 Turbo 码的译码与 RBF 神经网络的结合。本文提出的 RBF 神经网络译码算法只考虑了编码寄存器为 2 时的卷积 Turbo 码，对更为复杂的卷积 Turbo 码没有进行考虑。根据目前的仿真结果推测，对编码寄存器大的卷积 Turbo 码译码，译码会复杂很多，计算量也较大。

(2) 滑动窗口大小和卷积 Turbo 码的 RBF 神经网络译码算法复杂度的关系。目前的仿真结果表示，随着滑动窗口的增大，译码性能会越好，但译码复杂度会增大。在增大窗口，提高译码性能时，降低译码复杂度将是 RBF 神经网络译码的研究内容之一。

参考文献

[1] 王新梅, 肖国镇, *纠错码原理与方法*, 西安: 西安电子科技大学出版社, 2003年5月, Pages: 1-16, 378-379.

[2] 曹志刚, 钱亚生, *现代通信原理*, 北京: 清华大学出版, 2002 年 3 月, Pages:324-330.

[3] C. Berrou, A. Glavieux, and P. Thitimajshima, Near Shannon limit error-correcting coding and decoding: Turbo-codes (1), *IEEE Int. Conf. Communications ICC'93,* Geneva, Switzerland, May 1993, Volume 23, Pages: 1064–1071.

[4] C. Berrou and A. Glavieux, Near optimum error correcting coding and decoding: Turbo-codes, *IEEE Trans . Commun.*, Oct. 1996, Volume 44, Pages: 1261–1271.

[5] R. M. Pyndiah, Near-optimum decoding of product codes: Block turbo codes, *IEEE Transactions on Communications*, August 1998, 46(8): 1003-1010.

[6] S. Dave, Kim Junghwan, S. C. Kwatra, An efficient decoding algorithm for block turbo codes, *IEEE Transactions on Communications*, Jan. 2001, 49(1): 41-46.

[7] Z. Wang, K. K. Parhi, High performance ,high throughput turbo/SOVA decoder design, *IEEE Transactions on Communications*, April 2003, 51(4): 570-579.

[8] 王莹, 李晖, 王育民, 短帧Turbo码交织器的设计, *电路与系统学报*, 2004年8月, 9(4): 134-137.

[9] I. Sason, S. Shamai, On improved bounds on the decoding error probability of block codes over interleaved fading channels, with applications to turbo-like codes, *IEEE Transactions on Information Theory*, Sep. 2001, 47(6): 2275-2299.

[10] L. R. Bahl, J. Cocke, F. Jelinek, and J. Rativ, Optimal decoding of linear codes for minimizing symbol error rate, *IEEE Trans. Inform. Theory*, Mar. 1974, Volume IT-20, Pages: 284–287.

[11] B Bai, X Ma, X Wang, Novel algorithm for continuous decoding of turbo codes, *IEE Proc-Comm*, London, UK, Oct 2000, 146(5): 271-274.

[12] 林家儒, 吴伟陵, 冯志勇, Turbo 码译码算法在频率选择性信道中的修正, *北京邮电大学学报*, 2001 年 3 月, 24(1): 6-11.

[13] 李峰, 李建东, 无线 OFDM 系统频率选择性衰落信道下的 SOVA 译码算法, *通信学报*, 2004 年 4 月, 32(4): 544-547.

[14] 王东明, 高西奇, 赵春明, 尤肖虎, 空时分组码分块传输系统中的 Turbo 检测译码, *通信学报*, 2004 年 6 月, 25(6): 1-11.

[15] M. Tuchler, Design of Serially Concatenated Systems Depending On the Block Length, *IEEE Transactions on Communications*, Feb. 2004, 52(2): 209-218.

[16] Xiang Cheng, Dongfeng Yuan, Xiangming Yu, The improvement of turbo equalization through using turbo codes, *International Conference on Communications, Circuits and Systems*, Hong Kong , China, 27-30 May 2005, Volume 1, Pages: 124-127.

[17] Jaspar X., Vandendorpe L., New iterative decoding of variable length codes with turbo-codes, *IEEE International Conference on Communications*, Paris, France, 20-24 June 2004, Volume 5, Pages: 2606-2610.

[18] Talakoub S., Sabeti L., Shahrrava B., Ahmadi M., A linear log-MAP algorithm for turbo decoding and turbo equalization, *IEEE International Conference on Wireless And Mobile Computing*, Montreal, Canada, 22-24 Aug. 2005, Volume 1, Pages: 182-186.

[19] Divsalar D., Dolinar, S., Pollara F., Iterative turbo decoder analysis based on Gaussian density evolution, *21st Century Military Communications Conference Proceedings*, Los Angeles, California, 22-25 Oct.2000, Volume 1, Pages: 202-208.

[20]Yee Mong-Suan, B. L. Yeap, L. Hanzo, Radial basis function-assisted turbo equalization, *IEEE Transactions on Communications*, April 2003, 51(4): 664-675.

[21] Kuan Ee-Lin, Ng soon Xin, L. Hanzo, Joint-detection and interference cancellation based burst-by-burst adaptive CDMA schemes, *IEEE Transactions on Vehicular Technology*, Nov. 2002, 51(6): 1479-1493.

[22] M. Noorbakhsh, K. Mohamed-Pour, Combined turbo equalization and block turbo coded modulation, *IEE Proceedings-Communications*, London, UK, June

2003, 150(3): 149-152.

[23] 余小明, 周世东, 姚彦, OFDM传输中的子带自适应Turbo编码调制, *电子与信息学报*, 2004年8月, 26(8): 1312-1318.

[24] 左健存, 宋文涛, 徐友云, 罗汉文, 基于逐比特MAP算法的Turbo编码调制方式, *上海交通大学学报*, 2003年6月, 37(6): 812-815.

[25] 刘伟, 张海林, 刘增基, 差分检测 Turbo 乘积编码调制的研究, *西安电子科技大学学报(自然科学版)*, 2004年6月, 31(3): 362-366.

[26] H. Chen, X. Deng, A. Himovich, Layered Turbo Space-Time Coded MIMO-OFDM Systems for Time Varying Channels, *Globecom'03*, San Francisco , USA, 1-5 Dec. 2003, Volume 4, Pages: 1831-1836.

[27] 王亚峰, 杨鸿文, 杨大成, 采用Turbo码的typeIII HARQ性能分析, *通信学报*, 2004年6月, 25(6): 139-146.

[28] Caid William R., Means Robert W., Neural network error correcting decoders for block and convolutional codes, *IEEE Global Telecommunications Conference (GLOBECOM'90)*, San Diego, CA, Dec. 1990, Volume 2, Pages:1028 – 1031.

[29] El-Khamy Said E., Youssef El-Sayed A. and Abdou Hossam-El-Din M., Soft decision decoding of block codes using artificial neural network, *IEEE Symposium on Computers and Communications*, 27-29 June 1995, Pages: 234-240.

[30] Xiao-An Wang, Wicker S. B, An artificial neural net Viterbi decoder, *IEEE Transactions on Communications*, Feb. 1996, Volume 44, Issue 2, Pages: 165-171.

[31] Annauth R., Rughooputh H. C. S., Neural network decoding of turbo codes, *International joint conference on Neural Conference (IJCNN'99)*, Washington DC, USA, 10-16 July 1999, Volume 5, Pages: 3336-3341.

[32] Yang Xiao, Ying Zhao, Increasing user capacity by using multimodes turbo en-decoding for CDMA systems, *TENCON 2004*, Chiang Mai, Thailand, 21-24 Nov. 2004, Volume B, Pages: 465-468.

[33]Yanhui Tong,Yeap T.-H., Chouinard J.-Y., VHDL implementation of a turbo decoder with log-MAP-based iterative decoding, *IEEE Transactions on*

Instrumentation and Measurement, Aug. 2004, Volume 53, Issue 4, Pages: 1268-1278.

[34] Lee L.-N., Hammons A.R. Jr., Feng-Wen Sun, Eroz M., Application and standardization of turbo codes in third-generation high-speed wireless data services, *IEEE Transactions on Vehicular Technology*, Nov. 2000, Volume 49, Issue 6, Pages: 2198-2207.

[35] 刘东华, *Turbo 码原理与应用技术*, 北京:电子工业出版社, 2004 年 1 月, Pages: 68-100.

[36] 高隽, *人工神经网络原理及仿真实例*, 北京:机械工业出版社, 2003 年 7 月, Pages: 44-66.

[37] Kou-Hu Tzou, Dunham J G, Sliding block decoding of convolutional codes, *IEEE Transactions on Communications*, Sep. 1981, Volume 29, Issue 9, Pages: 1401-1403.

致 谢

首先，非常感谢我的导师刘星成副教授。在本文的研究工作中，刘老师非常关心我的学习和科研进展情况，悉心向我传授信道编码的相关知识，在我科研遇到困难的时候，提出启发性的建议，注重引导我的研究方法，这对我学术水平和研究能力的提高着莫大的帮助。

本文在刘老师的指导和启发下顺利完成，感谢刘老师在过去两年中对我的关心和培养。同时感谢我们实验室陈卓峰、朱帜、王康、程浩辉和周敬莹同学对我在科研上的帮助。感谢评审老师对我论文提出的宝贵而中肯的修改意见。感谢电子系的各位老师，他们讲授的知识使我获益匪浅，为我的课题研究打下了坚实的基础。

最后感谢我的父母一直以来对我的关心和帮助，使我能够顺利完成硕士研究生的学习阶段。

发表文章

[1] 王一波, 刘星成, 基于 RBF 神经网络的卷积码译码, *中山大学学报（自然科学版）*, 2006 年 5 月, Volume 45 Sup., Pages: 106-109. 该文于第十五届中国计算机学会网络与数据通信学术会议宣读, 广州, 2006 年 5 月.

原创性声明

本人郑重声明：所呈交的学位论文，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，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究作出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：王一波

日期：2006年5月8日