

Implementation of Turbo Code Simulation Based on Simulink

Ding Wang

Hangzhou Dianzi University, Hangzhou
Email: wd15858268005@126.com

Received: May 12th, 2014; revised: May 19th, 2014; accepted: May 26th, 2014

Copyright © 2014 by author and Hans Publishers Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY).
<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

This paper describes the basic structure of Turbo code scheme, and presents a Simulink model of a Turbo encoder and decoder. Turbo encoder concatenates two same sub-encoders in parallel through interleaver. The iteration decoder is based on APP (A Posteriori Probability) decoder block in Simulink, so the process of coding becomes convenient and vivid. Then, based on the simulation we made, some main factors (such as interleaver length, iteration times, decoding algorithm) which will affect the performance of Turbo codes are studied, and the corresponding emulation results and the analysis are given. It has some reference value to the actual system design.

Keywords

Turbo Code, Simulink, A Posteriori Probability Decoder

一种基于Simulink的Turbo码 仿真实现

王 丁

杭州电子科技大学, 杭州
Email: wd15858268005@126.com

收稿日期: 2014年5月12日; 修回日期: 2014年5月19日; 录用日期: 2014年5月26日

摘要

介绍了 Turbo 码的编解码原理，并且提出了一种完全基于 Simulink 模块的 Turbo 码仿真模型。编码器采用两个相同的分量编码器通过交织器并行级联而成。译码器采用迭代译码方式，由 Simulink 模型库中的最大后验概率译码模块(A Posteriori Probability Decoder)构成，极大的简化了译码器的复杂性。通过仿真，分析了迭代次数、交织长度、译码算法对译码性能的影响。对实际的系统设计有一定的参考价值。

关键词

Turbo 码, Simulink, 最大后验概率译码

1. 引言

在 1993 年的 ICC (国际通信会议) 上, C. Berrou 等人最先提出了 Turbo 码的概念[1]。在 Turbo 码的设计思想中, 将传统的卷积码与随机交织器十分巧妙的结合在了一起, 既实现了随机编码, 又完成了由短码构造长码的设计思想。同时, Turbo 码还采用了软输入软输出(SISO)的迭代译码方式来最大程度地逼近最大似然译码。由此, Turbo 码得以获得了接近 Shannon 极限[2]的良好性能。虽然至今 Turbo 码在数学上的机理还没有定论, 但是它的优越的性能及迭代译码算法的思想已经被人们所理解。

Turbo 码是一种非常复杂的信道编码方案, 对 Turbo 码的具体实现还没有一个清楚的度量, 因此需要使用计算机对系统进行仿真。目前的仿真大多基于编程的形式, 是一种复杂和重复的方式, 虽然有人采用 Simulink 进行仿真, 但是对于交织、子译码等比较复杂的模块, 仍需要重新编程, 非常复杂。如何实现一种简单的仿真模型, 以满足各种研究的需要, 具有重要的现实意义, 因此本文提出了一种基于 Simulink 的 Turbo 码仿真模型, 该仿真模型完全基于 Simulink 的自有模块, 无需自己编写模块程序, 简化了编译器的复杂性。通过仿真, 分析了迭代次数、交织长度、不同译码算法对 Turbo 码性能的影响。

2. Turbo 码编译码原理

2.1. Turbo 码编码

Turbo 码的并行级联卷积码(PCCC)结构[3]如图 1 所示, 该结构主要由两个分量编码器 RSC、交织器、复用与删余器组成。

Turbo 码编码过程: 输入信息序列复用为相同的三路, 第一路直接送入复用与删余器, 第二路送入分量编码器 1, 第三路送入交织器交织后再送入分量编码器 2 编码。两路编码器的输出以及第一路信号输入复用与删余器进行处理, 构成码字序列输出。

分量编码器的最佳选择是递归系统卷积码(RSC), 输入信息经 RSC 输出码字具有较高的汉明距离, 交织器的功能是置乱原数据的排列数据, 降低码字之间的相关性。两者的结合是实现 Turbo 码近似随机编码的关键。复用与删余器作用是删除编码器输出校验序列中的部分校验位, 以此提高编码速率, 然后再与未编码的信息序列复用重组, 由并转串最后输出。

2.2. Turbo 码译码

Turbo 码的基本译码结构[4]如图 2 所示。

Turbo 码编译码译码器由两个软输入/软输出(SISO)译码器 DEC1 和 DEC2 串行级联而成, 交织器与编码器中所使用的交织器相同。译码器 DEC1 对分量码 RSC1 进行最佳译码。产生关于信息序列中每一

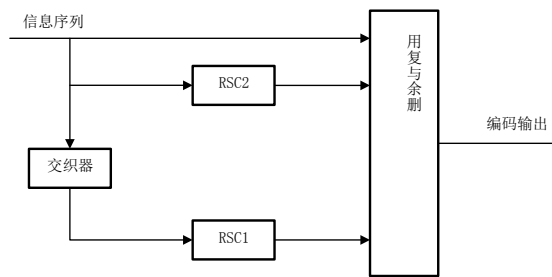


Figure 1. PCCC encoding structure
图 1. PCCC 编码结构

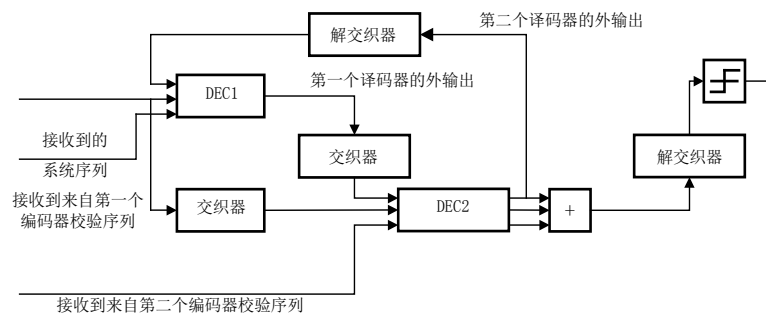


Figure 2. Decoding structure
图 2. 译码结构

比特的似然信息，并将其中的“新信息”经过交织送给 DEC2，译码器 DEC2 将此信息作为先验信息，对分量码 RSC2 进行最佳译码，产生关于交织后信息序列中每一比特的似然比信息，然后将其中的“外信息”经过解交织送给 DEC1，进行下一次译码。这样，经过多次迭代，DEC1 或 DEC2 的外输出信息趋于稳定，似然比值逼近于对整个码的最大似然译码，然后对此似然比进行硬判决，即可得到信息序列的最佳估计值。

3. Turbo 码的仿真设计

3.1. Turbo 码编码器设计

Turbo 码编码器采用两个相同的分量编码器通过交织器级联而成。分量编码器是码率为 $R=1/2$ 的 RSC 码，经过删余复用后总的 Turbo 码的码率 $R=1/3$ 。

序列产生用贝努利发生器(Bernoulli Binary Generator)。原始序列一路进入卷积编码器1(Convolutional Encoder1)，另一路经过随机交织器(Random Interleaver)后进入卷积编码器2(Convolutional Encoder2)。奇偶分离器1,2(Deinterlacer)将卷积编码器1,2的输出分别分成奇序列与偶序列。将卷积编码器1的奇序列、偶序列以及卷积编码器2的偶序列经复用器(Mux)合成一路串行序列，作为Turbo编码的输出。

卷积编码器的参数 $Trellis = poly2trellis(3,[7\ 5],7)$ ，由于存在约束与反馈，奇偶分离器1输出的偶序列与原始序列相同。

3.2. Turbo 码译码器设计

Simulink 提供的 APP Decode 模块完成卷积码的后验概率译码。可以用该模块构建 Turbo 译码器。他有两个输入端 $L(u)$ ， $L(c)$ 和两个输出端 $L(u)$ ， $L(c)$ 。输入端 $L(u)$ 输入原始序列的对数似然概率序列，输入端 $L(c)$ 输入编码器输出的对数似然概率序列。输出端 $L(u)$ ， $L(c)$ 是基于编码信息的更新序列的对

数似然概率序列。

编码器输出序列经高斯白噪声信道(AWGN)加噪后送往译码器。该信号经解复用器(Demux)分成三路信号。第 1、2 路信号作为奇偶连接器(interlacer1)的奇偶输入,合成一路信号 Out1。同理,第 1 路信号经随机交织器后与第 3 路信号输入奇偶连接器,合成一路信号 Out2。

Turbo 码译码器如图 3 所示,模块 APP Decode1 的输入端 $L(c)$ 的输入 In1 为信号 Out1,输入端 $L(u)$ 的输入为上一次迭代后的结果,输出端 $L(c)$ 终止,输出端 $L(u)$ 经交织后作为模块 APP Decode2 的输入端 $L(u)$ 的输入信号,Out2 信号作为该模块的输入端 $L(c)$ 输入信号 In2。模块 APP Decode2 的输出端 $L(c)$ 终止,而输出端 $L(u)$ 经解交织后得到原始序列的对数似然比,经过延迟单元(Delay)延迟一个迭代周期,输出将作为下次迭代的输入,这样就完成了一次迭代。经多次迭代后得到的原始序列的对数似然比将越来越逼近于整个码的最大似然译码,再经硬判决后可以得到原始序列的最佳估计值。

完成本次仿真的关键是如何控制迭代的次数。假设每次迭代需 t 秒,则可以让贝努利发生器的每一帧的采样时间为 t 秒,进行 n 次迭代需要 $n \times t$ 秒,通过零阶保持器(Zero Order Hold)将经过高斯白噪声信道的编码信号保持 $n \times t$ 秒,即可保证 n 次迭代。在第一次迭代时并没有上一次的迭代结果作为 APP Decode1 的输入端 $L(u)$ 的输入信号,为了避免这个错误,将迭代后信号与脉冲产生器(Discrete Pulse Generator)的输出信号相乘后作为 APP Decode1 输入端 $L(u)$ 的输入信号,脉冲产生器的参数设置为幅度为 1,周期为 $n \times t$ 秒,脉冲宽度为 $(n-1) \times t$,相位延迟为 t 秒,这样能保证第一次迭代时输入端有信号,信号为全 0 序列。

4. 仿真结果及分析

4.2. 迭代次数对 Turbo 码译码性能的影响

迭代译码结构是 Turbo 码具有良好的译码性能的重要原因。在交织长度为 1024,采用译码算法为 Max*时,选择迭代次数为 3, 7, 9, 对 Turbo 码性能影响进行比较。译码器性能如图 4 所示。

由图中可知,在相同的信噪比之下,迭代次数越多,误码率越低,译码性能越好。但是迭代系数为 7,9 时,译码性能差别甚微。这说明迭代次数为 7 时已经饱和[5],再增加迭代次数意义不大,反而增加不必要的计算负担。

4.2. 交织长度对 Turbo 码译码性能的影响

迭代次数为 7,译码算法为 Max*时,选取交织长度为 51,210,242,048 对 Turbo 码性能影响进行比较。

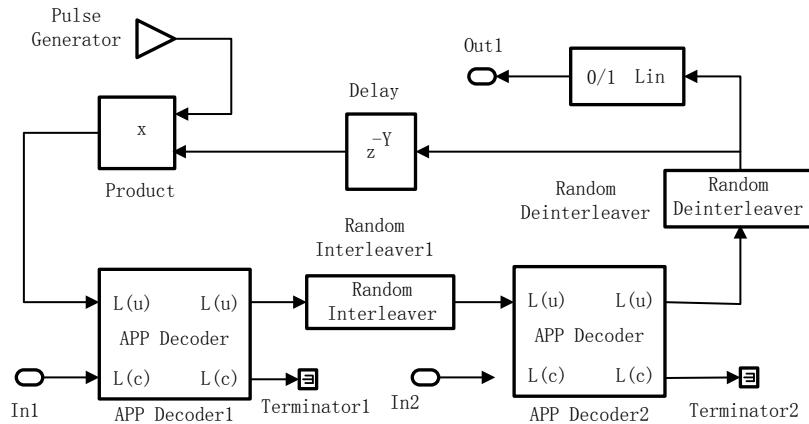


Figure 3. Decoder
图 3. 译码器

由图 5 可知,随着信噪比的提高, Turbo 码译码性能得到了显著的提高,而且交织长度越长,码译码性能越好。在实际的应用中,随着交织长度的增加,译码的复杂度,编码时延,传输时延,译码时延都会增加,所以需要根据具体的要求选定最佳的交织长度。

4.3. 译码算法对 Turbo 码译码性能的影响

APP 译码器可以选择三种译码算法: True App、Max*、Max。图 6 是这三种算法的译码性能的比较,其中交织长度为 1024,迭代次数为 7 次。比较而言,Max*算法与 APP 算法的译码性能相近,而且都优于 MAX 算法。

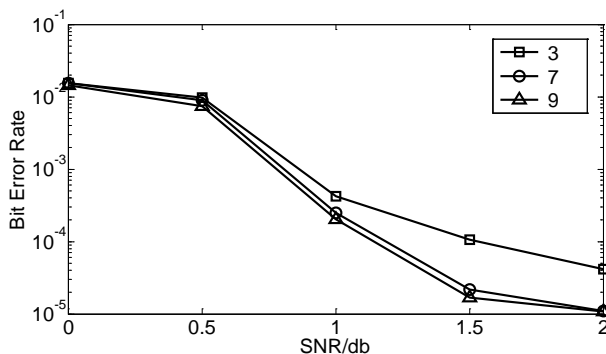


Figure 4. Decoding performance affected by iteration times

图 4. 迭代次数对译码性能的影响

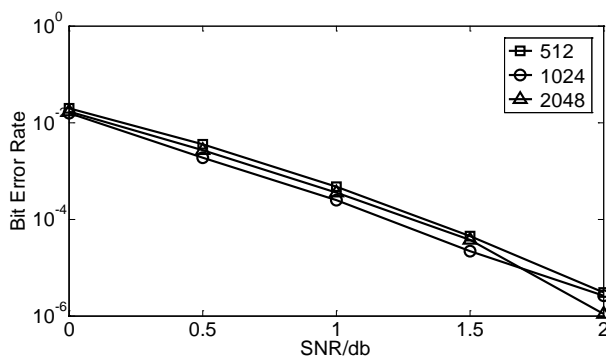


Figure 5. Decoding performance affected by interleaver length

图 5. 交织长度对译码性能的影响

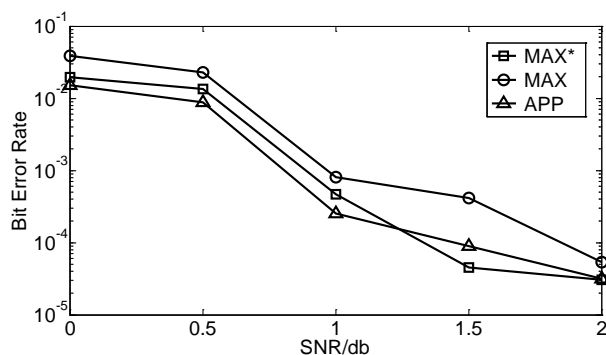


Figure 6. Decoding performance affected by decoding algorithm

图 6. 译码算法对译码性能的影响

5. 结束语

提出了一种完全基于 Simulink 模块的 Turbo 码仿真模型，编码器采用并行级联卷积码结构，译码器由最大后验概率译码模块构成迭代结构，可以选择三种译码算法，方便了 Turbo 码的分析与研究。通过仿真结果，分析了迭代次数、交织长度、不同算法对译码性能影响，对实际的系统设计有一定的参考价值。

参考文献 (References)

- [1] Vucetic, B. and Yuan, J.H., 著 (2004) 王晓海, 等, 译. 空时编码技术. 机械工业出版社, 北京.
- [2] 孙锴英 (2009) 无线通信中 Turbo 码技术研究及性能分析. 硕士论文, 山东大学, 济南.
- [3] 郑广宁 (2008) LTE 系统中 Turbo 编码方案研究. 硕士论文, 北京交通大学, 北京.
- [4] 周贤伟, 赵欣, 王丽娜 (2006) 使用 Simulink 构建 Turbo 码仿真系统. 微计算机信息, **15**, 202-204.
- [5] 邢莉, 王忠, 李兴国, 任昊 (2009) 基于 Matlab 的 Turbo 码仿真研究. 现代电子技术, **3**, 19-21.