

文章编号: 1001-2486 (2000) 05-0046-05

基于交叉熵最小化的 Turbo 码迭代译码停止准则*

刘东华, 唐朝京

(国防科技大学电子科学与工程学院, 湖南 长沙 410073)

摘要: 根据 Turbo 码最优译码算法及迭代译码的基本原理, 在保证迭代收敛的条件下, 利用交叉熵最小化原则推导出 Turbo 码译码过程中停止迭代的准则, 并给出了一种降低计算复杂性和减少存储空间的简化算法, 最后通过仿真证明了此迭代停止准则及其简化算法的有效性。

关键词: Turbo 码; 迭代译码; 交叉熵; 停止准则

中图分类号: TN911.22 **文献标识码:** A

Turbo Iterative Decoding Stop Criterion Based on Cross-entropy Minimization

LIU Dong-hua, TANG Chao-jing

(College of the Electronic Science and Engineering, National Univ. of Defense Technology, Changsha 410073, China)

Abstract: Based on the optimal decoding algorithm and basic principle of iterative decoding for turbo code, the stop criterion for the iterative decoding is deduced under the condition of cross-entropy minimization and iteration convergence. A simplified algorithm that reduces calculation complexity and store space comparison to the original criterion is worked out. Finally, it is proved that the iterative stop criterion and its simplified algorithm are both valid by simulation.

Key words: Turbo code; iterative decoding; cross-entropy; stop criterion

Turbo 码是 Berrou C 等人在 1993 年提出的一种新型信道编码方案^[1]。它由于很好地应用了香农信道编码定理中的随机性编译码条件而获得了几乎接近香农理论极限的译码性能。Turbo 码的最大特点是采用了迭代译码的思想。译码迭代次数直接影响计算复杂性和延时。在保证迭代收敛的前提下, 通过设计有效的停止准则, 可以适当减小译码迭代次数, 实现译码性能与复杂性及译码延时的协调。本文根据交叉熵最小化原理, 在分析 Turbo 码最优译码算法和迭代译码原理的基础上, 推导出了确定平均迭代次数的迭代译码停止准则, 并对其计算进行简化。通过仿真验证了应用此准则可以在使译码性能降低很小的情况下, 确定最小的译码迭代次数。

1 最优化交叉熵原理

交叉熵定义为两种不同概率分布之间的量度。对于离散的概率分布 $p(x)$ 和 $q(x)$, 交叉熵定义为^[2]:

$$H[p, q] = \sum_x p(x) \log \frac{p(x)}{q(x)} \quad (1)$$

交叉熵最小化是指: 给定一个先验分布 $q(x)$, 决定“最接近”(交叉熵意义下的)的另一个分布 $p(x)$, 使得 $p(x)$ 在一定的限制条件下, 满足

$$H[p, q] = \min_s H[s, q] \quad (2)$$

限制条件为

$$\sum_x p(x) = 1, E_p[f_i(x)] = 0 \quad i = 1, 2, \dots, N \quad (3)$$

其中, $f(\cdot)$ 代表对随机变量 x 的条件限制, $E_p[\cdot]$ 表示在 p 分布下对 $f(\cdot)$ 求平均值。

根据先验分布 $q(x)$, 可以得到满足最小交叉熵条件的概率分布

* 收稿日期: 1999-12-21

作者简介: 刘东华 (1976), 男, 博士生。

$$p(x) = q(x) \exp\left[-\lambda_0 \cdot \sum_{k=1}^N \mathcal{N}_k(x)\right] \quad (4)$$

其中 λ_0 和 λ_k 由式 (3) 决定。

2 Turbo 码的基本编译码原理

2.1 Turbo 码编码原理

图 1 给出了 Turbo 编码器的一般性结构。其中输入数据块 d , 经过交织器后送入 M 个成员码编码器 (一般选用递归系统卷积码) 进行成员编码的同时, 作为系统信息直接输出 (X_0)。成员编码得到校验输出序列 X_1, X_2, \dots, X_M , 这 M 个校验序列经压缩 (以得到不同的码率) 后与系统信息 X_0 一起经并/串变换并调制后发送出去。

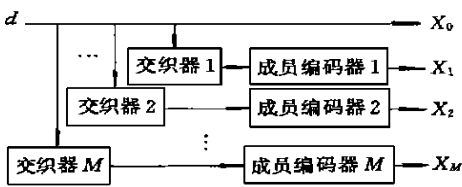


图 1 Turbo 码编码器结构

Fig. 1 Structure of turbo encoder

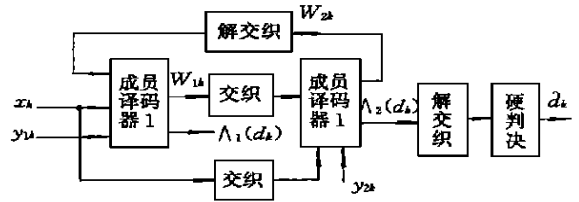


图 2 Turbo 码译码器结构

Fig. 2 Structure of turbo decoder

2.2 最优译码算法及译码流程

Turbo 码译码的最优算法是最大后验概率算法 (MAP)^[3]。以码率为 1/2 的二维 Turbo 码为例, 设 k 时刻的编码码字为 $C_k = (X_k, Y_k)$, 其中 X_k 为系统比特, Y_k 为校验比特。经 BPSK 调制后送入高斯加性白噪声无记忆 (AWGN) 信道, 接收端 k 时刻得到的接收码字为 $R_k = (x_k, y_k)$ 。接收信息经过译码预处理 (解调、并/串变换) 后送入译码器。对于输入信息比特 d_k , 其译码输出对数似然比 LLR 为

$$\Lambda(d_k) = \log \frac{p(d_k = 1 | R)}{p(d_k = 0 | R)} = \log \frac{\sum_m \sum_m p(d_k = 1, S_k = m, S_{k-1} = m', R_1^{k-1}, R_k, R_{k+1}^N)}{\sum_m \sum_m p(d_k = 0, S_k = m, S_{k-1} = m', R_1^{k-1}, R_k, R_{k+1}^N)} \quad (5)$$

其中 S_k 和 S_{k-1} 分别表示 k 和 $k-1$ 时刻的译码器状态, R_1^{k-1} 是从开始时刻到 $k-1$ 时刻的接收码字, R_k 为 k 时刻的接收码字, R_{k+1}^N 为 k 时刻以后的接收码字。

分别定义联合概率和条件概率 $\alpha_k^i(m)$, $\beta_k(m)$ 和 $\gamma_i(R_k, m, m')$:

$$\alpha_k^i(m) = p(d_k = i, S_k = m, R_1^k) \cdot p(d_k = i, S_k = m | R_1^k) / p(R_1^k) \quad (6)$$

$$\beta_k(m) = p(R_{k+1}^N | S_k = m) / p(R_{k+1}^N | R_1^k) \quad (7)$$

$$\gamma_i(R_k, m, m') = p(R_k | d_k = i, S_k = m, S_{k-1} = m') \cdot p(d_k = i | S_k = m, S_{k-1} = m') \cdot \pi(S_k = m | S_{k-1} = m') \quad (8)$$

由于 AWGN 信道上 k 时刻的接收码字与前面的接收码字无关, 由 Bayes 准则, 式 (5) 可以写成

$$\Lambda(d_k) = \log \frac{p(x_k | d_k = 1)}{p(x_k | d_k = 0)} + \log \frac{\sum_m \sum_m \gamma_1(y_k, m, m') \cdot \alpha_{k-1}(m') \cdot \beta_k(m)}{\sum_m \sum_m \gamma_0(y_k, m, m') \cdot \alpha_{k-1}(m') \cdot \beta_k(m)} \quad (9)$$

考虑到 x_k 是高斯分布的随机变量, 若记 W_k 为 $\Lambda(d_k) |_{x_k = \alpha}$ 则上式可写成

$$\Lambda(d_k) = L_c \cdot x_k + W_k \quad (10)$$

其中, $L_c = 4a \cdot E_s / N_0$ 为信道置信度, E_s / N_0 为信噪比。 W_k 与 x_k 不相关且与 d_k 符号相同, 因此把它作为先验信息送入译码器可以提高译码性能, 定义 W_k 为非本征信息。

Turbo 码的译码器结构如图 2 所示。对于每个被译比特 d_k , 成员译码器 1 (DEC1) 输出 $\Lambda_1(d_k)$ 及非本征信息 W_{1k} , W_{1k} 作为先验信息送入 DEC2, 相应地输出 $\Lambda_2(d_k)$ 和 W_{2k} 。 W_{2k} 通过反馈环送回 DEC1 的输入端, 作为 DEC1 的先验信息。若两个 DEC 均采用 MAP 算法, 则 DEC2 的输出似然比可写成

$$\Lambda_2(d_k) = f(\Lambda_1(d_k)) + W_{2k} \quad (11)$$

由于交织器和解交织器的作用, 非本征信息 W_k 与信道接收信息几乎不相关, W_{2k} 近似服从高斯分布^[3], 且其方差 $\sigma_W^2 \neq \sigma^2$, 由反馈译码, DEC1 的输出 $\Lambda_1(d_k)$ 为

$$\Lambda_1(d_k) = L_c \cdot x_k + 2W_{2k}/\sigma_W^2 + W_{1k} \quad (12)$$

最后, 由 DEC2 的输出 $\Lambda_2(d_k)$ 作硬判决, 得到译码输出

$$d_k = \text{sign}(\Lambda_2(d_k)) \quad (13)$$

3 迭代停止准则的推导及简化

3.1 基于交叉熵最小化的迭代停止准则

由 Turbo 码的迭代译码原理分析, 若译码算法是收敛的, 则随着译码迭代次数的增加, 误比特率应该持续减小^[4]。在第一次迭代译码后得到 DEC1 和 DEC2 的软输出 $\Lambda_1^{(1)}(d_k)$ 和 $\Lambda_2^{(1)}(d_k)$ (上标为迭代次数), DEC1 计算似然比时初始先验信息为 0, 而计算 $\Lambda_2^{(1)}(d_k)$ 时利用了 DEC1 输出的非本征信息 W_{1k} 作为先验信息, 根据式 (12) 并考虑到 W_{1k} 与被译比特同号, 所以硬判决输出时 $\Lambda_2^{(1)}(d_k)$ 要比 $\Lambda_1^{(1)}(d_k)$ 更接近正确的译码码字。在下一轮译码中, 计算 $\Lambda_1^{(2)}(d_k)$ 时利用了 DEC2 译码生成的非本征信息 W_{2k} 作为先验信息输入, 所以由 $\Lambda_1^{(2)}(d_k)$ 作硬判决输出的误比特率要比由 $\Lambda_2^{(1)}(d_k)$ 作硬判决输出的码字的误比特率小。在迭代译码收敛的条件下, 依次地利用 $\Lambda_1^{(i-1)}(d_k)$ 、 $\Lambda_2^{(i-1)}(d_k)$ 和 $\Lambda_1^{(i)}(d_k)$, ($i=2, 3, \dots$) 作为硬判决输入时译码误比特率依次减小。由此可见, 译码性能的提高是非本征信息作用的结果。但随着迭代次数的增加, 非本征信息与接收信息的相关性变大, 所能提供的有用信息减少, 从而对译码性能的影响降低。例如, 在信噪比 SNR=0.8dB, 若设第一次迭代得到的非本征信息的影响为 1, 则在 13 次迭代以后, 非本征信息的影响不超过 $0.04^{[3]}$, 而且随着 SNR 的加大, 非本征信息的影响也越来越小。随着译码次数的增加, $\Lambda_1^{(i)}(d_k)$ 和 $\Lambda_2^{(i)}(d_k)$ 的统计特性越来越接近。若干次迭代以后, 两个 DEC 的软输出服从相同的概率分布, 再进行迭代不会对译码性能产生任何影响 (只剩下 Turbo 码无法纠正的错误), 这时应停止迭代。而在实际的通信系统中, 考虑到译码复杂性和译码延时的影响, 迭代次数不可能太多, 因此考虑在达到误比特率要求时停止迭代。而由于在编、译码器中伪随机交织器的介入, 很难精确计算出所需的具体迭代次数。但由于随着迭代次数的增加, 两个 DEC 输出的似然函数的统计特性有“越来越接近”的趋势, 故可通过交叉熵来度量确定两者的接近程度, 并利用交叉熵最小化原理来近似确定译码迭代次数^[5]。

考虑两个成员译码器输出似然比的概率分布 $p_{DEC1}(d|R)$ 和 $p_{DEC2}(d|R)$ 及交叉熵定义, 有

$$\begin{aligned} H[p_{DEC1}(d|R), p_{DEC2}(d|R)] &= p_{DEC1}(d|R) \cdot \log \frac{p_{DEC1}(d|R)}{p_{DEC2}(d|R)} \\ &= \sum_{k=1}^N p_{DEC1}(d_k|R) \cdot \log \frac{p_{DEC1}(d_k|R)}{p_{DEC2}(d_k|R)} \end{aligned} \quad (14)$$

第二个等号是因为在 AWGN 信道上输出信息是相互独立的, 其中 N 为信息块的大小。

根据式 (12), 重新定义变量: 以 $\Lambda_1^{(i)}(d_k)$ 和 $\Lambda_2^{(i)}(d_k)$ 表示两个 DEC 对输入信息 d_k 第 i 次迭代译码后的输出似然比, $\sigma_{W_1}^{(i)}$ 和 $\sigma_{W_2}^{(i)}$ 表示第 i 次迭代后 DEC 输出非本征信息的方差, $W_{1k}^{(i)}$ 和 $W_{2k}^{(i)}$ 表示第 i 次迭代后输出的非本征信息, 于是有

$$\Lambda_1^{(i)}(d_k) = L_c \cdot x_k + \frac{2}{\sigma_{W_2}^{(i-1)}} \cdot W_{2k}^{(i-1)} + W_{1k}^{(i)}, \quad \Lambda_2^{(i)}(d_k) = L_c \cdot x_k + \frac{2}{\sigma_{W_1}^{(i-1)}} \cdot W_{1k}^{(i)} + W_{2k}^{(i)} \quad (15)$$

上式中, 非本征信息的方差可根据输入数据比特译码的统计信息 (二阶矩) 来计算

$$\sigma_W^2 = \frac{1}{N} \sum_{k=1}^N \left[|W_k| - \frac{1}{N} \sum_{k=1}^N |W_k| \right]^2 \quad (16)$$

由式 (5) 的似然比表示式, 有

$$\begin{aligned} p(d_k = 1 | R) &= \frac{\exp(\Lambda(d_k))}{1 + \exp(\Lambda(d_k))} = \frac{1}{1 + \exp(-\Lambda(d_k))}, \\ p(d_k = 0 | R) &= \frac{1}{1 + \exp(\Lambda(d_k))} = \frac{\exp(-\Lambda(d_k))}{1 + \exp(-\Lambda(d_k))} \end{aligned} \quad (17)$$

代入式 (14), 得到第 i 次迭代后两个成员译码器输出的交叉熵

$$\begin{aligned} H[p_{\text{DEC2}}(d | R), p_{\text{DEC1}}(d | R)] &= E_{p_{\text{DECI}}} \left\{ \sum_{k=1}^N \log \frac{p_{\text{DEC2}}(d_k)}{p_{\text{DECI}}(d_k)} \right\} \\ &= p_{\text{DEC2}}(d_k = 1) \cdot \log \frac{p_{\text{DEC2}}(d_k = 1)}{p_{\text{DECI}}(d_k = 1)} + p_{\text{DEC2}}(d_k = 0) \cdot \log \frac{p_{\text{DEC2}}(d_k = 0)}{p_{\text{DECI}}(d_k = 0)} \\ &= \sum_{k=1}^N \left[\frac{\Lambda_1^{(i)}(d_k) - \Lambda_2^{(i)}(d_k)}{1 + \exp(\Lambda_2^{(i)}(d_k))} + \log \frac{1 + \exp(-\Lambda_1^{(i)}(d_k))}{1 + \exp(-\Lambda_2^{(i)}(d_k))} \right] \end{aligned} \quad (18)$$

由式 (18) 可以看出: 随着迭代次数的增加, $\Lambda_1^{(i)}(d_k)$ 和 $\Lambda_2^{(i)}(d_k)$ 的差别减小, 从而 $p_{\text{DECI}}(d | R)$ 和 $p_{\text{DEC2}}(d | R)$ 的交叉熵减小; 当交叉熵达到最小值时, 就实现了最佳译码。但考虑实际系统的限制, 可根据具体系统对译码性能的要求和信道条件, 把交叉熵与给定门限相比较。当交叉熵小于门限时, 停止迭代。其中门限的选择与译码误比特率和信道条件有关。若记第 i 次迭代后两个成员译码器输出似然比的交叉熵为 $T(i)$, 迭代停止门限为 $\text{Th}(P_b(e), \text{SNR})$ 。则得到的停止迭代的准则为

$$T(i) < \text{Th}(P_b(e), \text{SNR}) \quad (19)$$

利用交叉熵最小化实现迭代停止准则, 迭代次数就变成了一个随机变量。在一定 SNR 条件和误比特率要求的前提下, 此随机变量的均值比要求的最大迭代次数要小, 而且随着 SNR 的增大, 其均值趋于稳定值。一般门限的取值为 $(10^{-2} \sim 10^{-4}) \cdot a \cdot T(1)$ 。其中 a 为调整参数, 取值根据不同系统要求确定。

在实际应用中, 根据上述准则得到平均迭代次数, 然后向正方向取整, 得到译码迭代次数。由此可以在译码性能降低不是很多的情况 (降低程度很小) 下, 实现译码复杂性及译码延时与译码性能的协调。

3. 2 算法简化

由式 (18) 可以看出, 计算 $T(i)$ 的过程中涉及到实数运算以及对数 (指数) 操作, 而且计算过程中需要存储两个 DEC 输出的似然值, 当数据帧较大时, 所需计算量和存储空间都是很大的。考虑到最后的译码输出是对似然比的硬判决, 故当第 i 次迭代后对于每个被译比特输出的似然比与第 $i-1$ 次迭代后输出的似然比同号时, 再进行迭代也不可能进一步提高译码性能 (所发生的译码错误是不可纠的)。比较相邻两次迭代输出似然比的符号, 若对帧内所有被译比特, 其似然比的符号均相同时, 则停止迭代; 否则进行下一轮迭代译码。这样, 相应的迭代停止准则变为

$$T(i) = H[p_{\text{DEC2}}^{(i-1)}, p_{\text{DEC2}}^{(i)}] = \sum_{k=1}^N \left[\frac{\Lambda_1^{(i-1)}(d_k) - \Lambda_2^{(i)}(d_k)}{1 + \exp(\Lambda_2^{(i)}(d_k))} + \log \frac{1 + \exp(-\Lambda_1^{(i-1)}(d_k))}{1 + \exp(-\Lambda_2^{(i)}(d_k))} \right] \quad (20)$$

当 DEC2 的第 i 次迭代后的输出 $\{\Lambda_2^{(i)}(d_k)\}$ 与第 $i-1$ 次迭代后的输出 $\{\Lambda_2^{(i-1)}(d_k)\}$ 同号时, 式 (20) 对于停止迭代来说是足够小的。简化算法仅需要做整数操作 (硬判决) 和比较操作, 且只需存储前一次迭代输出的硬判决值, 所以需要的计算量和存储空间都相对要小。

3. 3 仿真验证

在个人通信中常用的话音标准率为 9.6 Kbit/s, 相应的数据帧长为 192 比特, 我们对此情况作了仿真。Turbo 编码采用两个相同的递归系统卷积码由伪随机交织器并行级联的形式, 编码生成矩阵为 $(7, 5)_8$; 编码速率为 1/2; 信道模型为 AWGN 信道; 译码算法是 MAP 算法; 传输信息帧为 1500 帧; 最大译码迭代次数为 8 次。门限取作 $5 \times 10^{-4} \cdot T(1)$ 。得到的仿真结果如表 1 所示。可见, 在译码性能降低不是很多时, 采用上述迭代停止准则可以减少必要的译码迭代次数。而且用简化算法作为停止条件得到的平均迭代次数以及译码性能的降低, 与原始停止准则时基本上一致。

表1 停止准则下平均迭代次数及其对译码性能的影响

Tab. 1 Average iterative number under stop criterion and its influence to turbo decoding performance

信噪比 (dB)	8次迭代后 的误比特率 ($P_e(b)$)	停止准则下的性能			简化算法下的性能		
		平均迭 代次数	误比特率 ($P_e(b)$)	性能降低 程度	平均迭 代次数	误比特率 ($P_e(b)$)	性能降低 程度
1. 0	2.76×10^{-2}	6. 83	2.91×10^{-2}	6. 16%	6. 64	2.96×10^{-2}	7. 37%
2. 0	1.59×10^{-3}	5. 57	1.68×10^{-3}	5. 43%	5. 02	1.69×10^{-3}	6. 11%
3. 0	1.25×10^{-4}	3. 68	1.31×10^{-4}	5. 08%	3. 57	1.33×10^{-4}	5. 74%

4 结束语

Turbo 码在高噪声环境下超乎寻常的译码性能使得它在许多通信系统中有很大的应用潜力, 但迭代译码的计算复杂度和延时大大限制了它的应用。本文所提出的迭代停止准则可以针对不同的系统要求来确定译码的迭代次数, 从而减小计算量和延时。简化算法使准则更具有实用性, 可以把此准则作为一项指标来衡量在具体的通信系统中是否可以采用 Turbo 码作为其信道编码方案。

参考文献:

- [1] Berrou C, Glavieux A, Thitimajshima P. Near Shannon limit error correcting coding and decoding: turbo_codes (1) [C]. In ICC 93 Geneva, Switzerland May, 1993: 1064~ 1070.
- [2] Shore J E, Johnson W. Properties of cross-entropy minimization [J]. IEEE Transactions on Information Theory July, 1981: 472~ 482.
- [3] Berrou C, Glavieux A. Near Optimum Error Correcting Coding and Decoding: Turbo-codes [J]. IEEE Transactions on Communications Vol. 44, Oct 1997: 1261~ 1271.
- [4] Benedetto S, Divsalar D, Montorsi G, Pollara F. Soft- output decoding algorithms in iterative decoding of turbo codes [R]. TDA Progress Report 42~ 119 February, 1996: 63~ 87.
- [5] Moher M. Decoding via cross-entropy minimization [C]. IEEE Globecom Conference of 1993 Houston, TX, December, 1993: 809~ 813.