

LDPC 码的一种低复杂度 BP 译码算法

郭锐^{1,2}, 刘济林¹

(1. 浙江大学 信息与电子工程学系, 浙江 杭州 310027; 2. 杭州电子科技大学 通信工程学院, 浙江 杭州 310018)

摘要: 针对低密度奇偶校验(LDPC)码的BP译码算法在每一次迭代过程中,都要对全部比特和校验信息进行更新,存在计算量大、译码效率低的问题,提出了一种改进的BP译码算法.由于不同的比特节点和校验节点,其可靠程度不同,对BP译码的贡献也不同,为此给出了一种新的可靠性判断准则:采用每个比特的非法校验数和每次迭代过程中比特的伪后验概率的差来判断比特的可靠性,认为非法校验数小、伪后验概率差大的节点具有较高的可靠度.对可靠性较高的比特,下一次迭代过程中不参与更新,只更新那些有可能发生错误的比特.仿真结果表明,改进的BP译码算法在损失极少译码性能的情况下,大大地减少了迭代过程中的计算量,提高了译码效率.

关键词: 低密度奇偶校验; 置信传播算法(BP算法); 译码效率

中图分类号: TN91

文献标识码: A

文章编号: 1008-973X(2008)03-0450-06

New low complexity belief propagation decoding of low density parity check codes

GUO Rui^{1,2}, LIU Jilin¹

(1. Department of Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China;

2. College of Communication Engineering, Hangzhou Dianzi University, Hangzhou 310037, China)

Abstract: An improved belief propagation (BP) decoding scheme was put forward to solve the high complexity and low decoding efficiency problem of BP in low density parity check (LDPC) codes. Because different bit nodes and check nodes have different reliability and contribute differently to BP decoding; the proposed decoding scheme employs a differentiate rule based on lawless check number and pseudo posterior probability to evaluate every bit's reliability before each step of iteration; The nodes have little lawless check number and large pseudo posterior probability are supposed to have higher reliability. The bits with high reliability need not to be updated by the decoder. Simulation shows that the proposed BP decoding algorithm can reduce the computational load and increase the decoding efficiency significantly.

Key words: low density parity check (LDPC); belief propagation (BP) algorithm; decoding efficiency

由 Gallager^[1]首先提出的低密度奇偶校验(LDPC)码是一种线性分组码,其校验矩阵是稀疏矩阵,即矩阵中大部分元素均为零. Mackay^[2]从理论和实际应用两个方面详细论述了LDPC码的编、译码问题,提出了实用的概率迭代译码方法——置信传播算法(BP algorithm),极大地推动了LDPC码的发展. LDPC码因具有描述简单、性能优良、实

现复杂度低的特点,近年来受到越来越多研究者的关注,在移动通信领域得到广泛应用^[3].

虽然相对于Turbo码来说,LDPC译码的复杂性已经很低.然而,BP译码算法存在以下问题:(1)在每次迭代过程中,大量使用了浮点型运算,尤其是采用了很多的浮点型对数和指数运算;(2)由于在每一次迭代过程中,都要对全部比特和校验的信息

收稿日期: 2006-08-26.

浙江大学学报(工学版)网址: www.journals.zju.edu.cn/eng

基金项目: 国家自然科学基金重大资助项目(60534070); 浙江省科技厅重点资助项目(2005C14008).

作者简介: 郭锐(1980-),男,湖北十堰人,博士生,从事通信信号处理研究. 通讯联系人: 刘济林,男,教授. E-mail: Liuji@zju.edu.cn

进行计算. LDPC 码 BP 译码算法还是相当复杂的. 为了降低 BP 译码算法的复杂度, 一些学者做了大量的工作. Fossorier 等人^[4-5]提出了 UMP(uniformly most powerful)-BP 和归一化 BP 算法, 很好地控制了浮点型对数和指数运算, 解决了问题(1). Hu 等人^[6]通过研究 BP 算法的特点, 提出了 BP 算法的两种实现结构来提高 BP 算法的速度. Mao 等人^[7]对标准的 BP 的算法作了进一步的改进, 提出了一种称为 probabilistic 的新的传递机制. Zhang 等人^[8]提出了 Shuffled BP 算法, 通过尽早利用已更新节点的信息, 加快 BP 算法的收敛速度. 但针对问题(2)的研究却很少报道. 实际上在 BP 译码算法中, 大部分的错误信息都是在最开始的几次迭代中修正的, 而后面的迭代过程只纠正极少量的错误信息, 但每次迭代的计算量是完全一样的. 因此在译码过程中, 每一次迭代的效率是不同的, 前几次迭代的译码效率较高, 随着迭代次数的增加, 译码效率越来越低.

本文提出了一种低复杂度、高效率地 BP 译码算法, 有效地克服了在每次迭代过程中, 都要对全部的比特和校验信息进行更新的问题, 大大地降低了每次迭代过程中的计算量, 提高了译码效率和收敛速度.

1 LDPC 编码概述

LDPC 码是一种线性分组码, 它的校验矩阵 H 是稀疏矩阵. 用 $(n; j; k)$ 来表示一个码长为 n 、校验矩阵的每列和每行中分别有 j 和 k 个 1 的 LDPC 码. 按照在 LDPC 码校验矩阵中列的码重相等与否, LDPC 码可以分为规则 LDPC 码和非规则 LDPC 码. 每一个 LDPC 码的奇偶检验矩阵 H 都对应一个双向图, 在双向图上比特点 (bit node) 表示校验矩阵的列, 校验点 (check node) 代表校验矩阵的行. 一个 $n=10, j=2, k=4$ 的规则 LDPC 码的例子如图 1.

$$H= \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} . \tag{1}$$

非规则 LDPC 码与规则 LDPC 码相比有更强的纠错性能^[9-11]. 对于非规则 LDPC 码而言, 在奇偶检验矩阵中, 各行各列的非零元素个数不要求相等, 也就是说比特点和校验点的度数不是固定不变的.

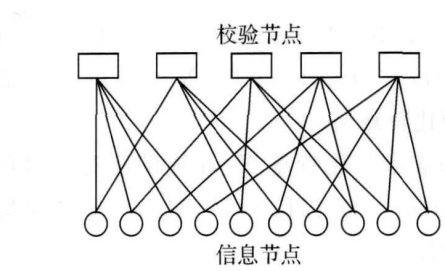


图 1 LDPC 码双向图

Fig. 1 Parity-check bipartite graph

非规则 LDPC 码的列及行的码重分布可以用如下两个多项式来表示^[12]:

$$\lambda(x) = \sum_{i=1}^{d_v} \lambda_i x^{i-1}, \tag{2}$$

$$\rho(x) = \sum_{i=1}^{d_r} \rho_i x^{i-1}. \tag{3}$$

相应的码率为

$$R = 1 - \sum_i \frac{\rho_i}{i} \setminus \sum_i \frac{\lambda_i}{i}. \tag{4}$$

式中: λ_i 和 ρ_i 分别为从比特点和校验点出发的度为 i 的边所占的百分比; d_v 和 d_r 分别为比特点和校验点中的最大度数. 如果比特点的个数是 N , 校验点的个数是 M , 则在双向图上度数为 i 的比特点和校验点的个数分别为

$$\left. \begin{aligned} N \frac{\lambda_i/i}{\sum_{j=2}^{d_v} \lambda_j/j} &= N \frac{\lambda_i/i}{\int_0^1 \lambda(x) dx}, \\ M \frac{\rho_i/i}{\sum_{j=2}^{d_r} \rho_j/j} &= M \frac{\rho_i/i}{\int_0^1 \rho(x) dx} \end{aligned} \right\} \tag{5}$$

2 LDPC 的传统的 BP 译码算法

图 1 给出了 LDPC 码校验矩阵以及所对应的双向图. 从图中可以看出, 每个比特点只和校验点直接相连, 而每个校验点也只和比特点直接相连. 双向图中各点之间并不是孤立的, 当译码器接收到一个码字时, 根据接收到的可靠性信息, 可以获得每一比特的可靠性程度. 然而, 由于每一个校验点在双向图中都有与之相连的数个比特点, 根据这些比特点的可靠性信息, 就可以计算出每一个校验点的可靠性程度. 根据校验点的可靠性程度, 又可以更新比特点的可靠性信息. 反复迭代这两个类点之间的可靠信息, 就得到的 BP 译码算法. 为了便于描述, 对 BP 译码算法中涉及的各符号的含义加以说明.

$N(m)$ 表示与校验点相连的所有比特点集合, 即 $N(m) = \{x_n: H_{mn} = 1\}$, $N(m) \setminus m$ 表示集合 $M(n)$ 去掉比特点 s_m .

$M(n)$ 表示与比特点 X_n 相连的所有校验点的集合, 即 $M(n) = \{s_m: H_{mn} = 1\}$, $M(n) \setminus m$ 表示集合 $M(n)$ 去掉校验点 s_m .

q_{mn}^x 表示基于接收信号并根据校验点集合 $M(n) \setminus m$ 的信息而得到的比特 $x_n = x$ 的概率, 其中, $x \in \{0, 1\}$. 而且 $q_{mn}^0 + q_{mn}^1 = 1$, 可以认为 q_{mn}^x 是比特点 n 向校验点 m 传递的信息.

r_{mn}^x 表示当比特 $x_n = x$, 且 $M(n) \setminus m$ 中的比特的概率为 $q_{mn'}^{x'}$ ($n' \in M(n) \setminus m$) 时, 校验点 m 对应的校验方程成立的概率. r_{mn}^x 可以看成是校验点 m 向比特点 n 传递的信息.

完整的 BP 算法可为, 对满足 $H_{mn} = 1$ 的 m, n 执行如下步骤:

1) 初始化

设 $p_n^0 = P(x_n = 0)$ ($x_n = 0$ 的先验概率), $p_n^1 = P(x_n = 1) = 1 - p_n^0$, 则 $q_{mn}^0 = p_n^0, q_{mn}^1 = p_n^1$.

2) 校验点更新, 令

$$\delta q_{mn} = q_{mn}^0 - q_{mn}^1, \delta r_{mn} = \prod_{n' \in N(m) \setminus n} \delta q_{mn'},$$
$$r_{mn}^0 = 0.5(1 + \delta r_{mn}), r_{mn}^1 = 0.5(1 - \delta r_{mn}).$$

3) 比特点更新

$$q_{mn}^0 = \alpha_{mn} p_n^0 \prod_{m' \in M(n) \setminus m} r_{mn'}^0, q_{mn}^1 = \alpha_{mn} p_n^1 \prod_{m' \in M(n) \setminus m} r_{mn'}^1.$$
其中选择 α_{mn} , 使得 $q_{mn}^0 + q_{mn}^1 = 1$.

4) 更新伪后验概率 q_n^0 和 q_n^1

$$q_n^0 = \alpha_n p_n^0 \prod_{m \in M(n)} r_{mn}^0, q_n^1 = \alpha_n p_n^1 \prod_{m \in M(n)} r_{mn}^1.$$
其中选择 α_n , 使得 $q_n^0 + q_n^1 = 1$.

5) 比特判决

若 $q_n^0 > 0.5$, x_n 判决为 0; 否则, x_n 判决为 1, $n = 1, 2, \dots, N$. 若 $H^T x = 0$ 或者迭代次数到达大迭代次数, 则结束, x 作为译码输出; 否则转步骤 2) 继续迭代.

3 改进的 BP 译码算法

尽管 BP 算法性能优良, 但它的运算量还是很大, 译码效率低, 表 1 给出了 $(N, J, 2J)$ LDPC 码在 BP 译码算法中, 每次迭代所需的计算量^[4]. 而且在每次迭代过程中, 都要对全部比特和校验信息进行计算, 所以每次计算量是完全相同的, 但是随着迭代次数的增加, 每次迭代过程中纠正的比特数越来越少. 图 2 给出了规则 LDPC 码 (4 096, 3, 6) 在

$E_b/N_0 = 1.6$ dB 时 BP 算法每一次迭代所纠正的错误比特占总错误比特的百分比. 可以看出, 只有前两次迭代纠正的比特数超过了总错误比特数的 10%, 其他次迭代纠正的比特数有的甚至不足 1%. 但这些迭代计算量却完全相同. 因此随着迭代次数的增加, 译码效率越来越低. 译码效率可表示为

$$\eta = B/C. \tag{6}$$

式中: η 为译码效率, B 为每次迭代过程中纠正的比特数, C 为每次迭代的运算量.

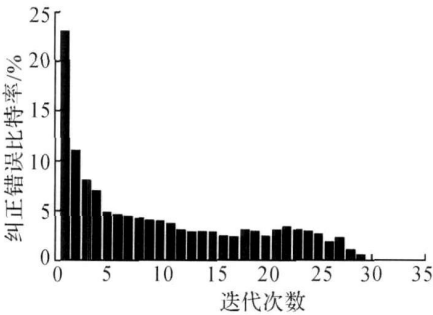


图 2 传统 BP 算法每次迭代过程中纠正错误比特率

Fig. 2 Corrected error bit in every iteration of traditional BP decoding

表 1 $(N, J, 2J)$ LDPC 码 BP 译码算法所需计算量

Tab. 1 Computational load in each iteration of $(N, J, 2J)$ LDPC code

概率	次数	类型
δq_{mn}	NJ	浮点加
δr_{mn}	$3N(J-1)$	浮点乘
r_{mn}^x	NJ	浮点加
$\prod r_{mn}^x$	$2N(3J-4)$	浮点乘
α_{mn}	NJ	浮点除
q_{mn}^x	NJ	浮点乘
α_n	N	浮点除
q_{mn}^x	$2N$	浮点乘

本文提出了一种低复杂度、高效率的 BP 译码算法, 采用每个比特点的非校验数结合比特点伪后验概率的差来判断比特可靠性, 通过判决准则让一部分节点提前终止迭代, 减少迭代过程中不必要的更新从而降低计算量、提高译码效率. 具体描述如下:

3.1 非法校验数 $U(l)$ 的求解

在一个 (n, j, k) LDPC 码中, 一共存在 nj/k 个校验集合, 每个集合由 k 个比特组成, 每个比特同时属于 j 个校验集合. 若一个校验集合中的 k 个比特的模 2 和为 0, 称该校验为一个合法校验, 反之, 称其为一个非法校验.

若用 G 和 H 来分别表示一个 LDPC 码的生成阵和校验阵, 那么满足

$$(\mathbf{H} \cdot \mathbf{t}) \bmod 2 = 0 \quad (7)$$

的 \mathbf{t} 均为合法的码字. 当一个信息序列 \mathbf{s} 经过生成阵 \mathbf{G} 作用编码后通过加性噪声信道, 那么在接收端收到的信号可以表示为

$$\mathbf{r} = (\mathbf{G}^T \mathbf{s} + \mathbf{n}) \bmod 2. \quad (8)$$

\mathbf{n} 表示信道加性噪声. 由于 $\mathbf{t} = (\mathbf{G}^T \mathbf{s}) \bmod 2$, 在接收端有

$$\begin{aligned} (\mathbf{H} \cdot \mathbf{r}) \bmod 2 &= [\mathbf{H} \cdot (\mathbf{G}^T \mathbf{s} + \mathbf{n}) \bmod 2] \bmod 2 = \\ &= [\mathbf{H} \cdot (\mathbf{t} + \mathbf{n}) \bmod 2] \bmod 2 = [\mathbf{H} \cdot \mathbf{n}] \bmod 2 = \mathbf{Z}. \end{aligned} \quad (9)$$

可以看出, 向量 \mathbf{Z} 中的每一个非零项都代表了一个非法校验.

因此在接收端, 译码器可用下式来计算每个比特的非法校验数目:

$$\mathbf{U}^T \cdot [(\mathbf{H} \cdot \mathbf{r})^T] \bmod 2 \cdot \mathbf{H} = \mathbf{Z}^T \cdot \mathbf{H}. \quad (10)$$

式中: 列向量 \mathbf{U} 的每一个元素对应每一个比特的非法校验数. 用 $U(l)$ 表示比特 l 的非法校验数.

3.2 伪后验概率提供的比特可靠信息

在获取了每一比特的非法校验后, 译码器可以根据非法校验数来判断比特的可靠程度. 可以认为 $U(l)$ 为 0 的比特(这些比特没有非法校验)可靠性程度比较高. 然而, 仅仅 $U(l) = 0$ 并不能保证比特点 l 的绝对可靠. 由于与 l 连接的校验点, 以及相邻的比特点的可靠性不能得到保证. 在 BP 译码算法中, 每次比特点更新后, 都要计算两个伪后验概率 q_n^0 和 q_n^1 . 本文采用伪后验概率来辅助判断.

令 $S(l)$ 表示接收到的伪后验概率提供的比特 l 可靠性信息. $S(l)$ 定义如下:

$$S(l) = |q_n^0 - q_n^1|; 0 \leq S(l) \leq 1. \quad (11)$$

当 $U(l) = 0$ 且 $S(l)$ 取值比较大时, 认为比特 l 的可靠程度比较高. 最严格的门限是 $U(l) = 0, S(l) = 1$, 其表示不更新满足 $U(l) = 0, S(l) = 1$ 的比特的信息. 这是最为严格的情况, 因为 $U(l) = 0$ 意味着这个比特没有非法校验, 而 $S(l) = 1$ 意味着接收到比特为 1 或 0 的概率为 1. 实际上在迭代中, 尤其是在开始迭代的时候, 这个条件太严格. 因此通常取一个 $U(l) = 0, S(l) = M$ (M 是一个接近 1 的数, 比如 0.8) 作为门限. 用 Φ 表示更新过程中不需要迭代的比特集合.

3.3 改进 BP 算法的实现

通过以上的分析, 改进的 BP 算法修正如下:

1) 初始化

设 $p_n^0 = P(x_n = 0)$ ($x_n = 0$ 的先验概率), $p_n^1 = P(x_n = 1) = 1 - p_n^0$, 则

$$q_{mn}^0 = p_n^0, q_{mn}^1 = p_n^1, \Phi = \emptyset.$$

2) 计算每一比特的非法校验数目 $U(l)$

$$\mathbf{U}^T \cdot [(\mathbf{H} \cdot \mathbf{r})^T] \bmod 2 \cdot \mathbf{H} = \mathbf{Z}^T \cdot \mathbf{H}.$$

3) 校验点更新, 令 $\delta q_{mn} = q_{mn}^0 - q_{mn}^1$

$$\delta r_{mn} = \prod_{n' \in N(m) \setminus n \text{ 且 } n' \notin \Phi} \delta q_{mn'},$$

$$r_{mn}^0 = 0.5(1 + \delta r_{mn}), r_{mn}^1 = 0.5(1 - \delta r_{mn}).$$

4) 比特点更新

$$q_{mn}^0 = \alpha_{mn} p_n^0 \prod_{m' \in M(n) \setminus m} r_{m'n}^0, q_{mn}^1 = \alpha_{mn} p_n^1 \prod_{m' \in M(n) \setminus m} r_{m'n}^1,$$

其中选择 α_{mn} , 使得 $q_{mn}^0 + q_{mn}^1 = 1$.

5) 更新伪后验概率 q_n^0 和 q_n^1

$$q_n^0 = \alpha_n p_n^0 \prod_{m \in M(n)} r_{mn}^0, q_n^1 = \alpha_n p_n^1 \prod_{m \in M(n)} r_{mn}^1.$$

6) 计算接收到的 l 的后验概率提供的可靠性信息为

$$S(l) = |q_n^0 - q_n^1|.$$

7) 记录下次迭代过程中不需要更新的比特 l , l 满足 $U(l) = 0, S(l) > M$, 则把 l 加入集合 Φ .

8) 比特判决

若 $q_n^0 > 0.5, x_n$ 判决为 0; 否则, x_n 判决为 1, $n = 1, 2, \dots, N$. 若 $\mathbf{H}^T \mathbf{x} = 0$ 或者迭代次数到达大迭代次数, 则结束, \mathbf{x} 作为译码输出; 否则转步骤 2) 继续迭代.

3.4 改进 BP 算法在对数域的实现

以上表述的 BP 算法在每个校验点或者比特点分别计算各个比特处于 0 或者 1 的概率, 并且在计算 q_{mn}^x 和 q_n^x 要选择合适的系数 α_{mn} 和 α_n 使之满足条件, 算法在表述上不是十分的简明. 采用对数似然比后, BP 算法有一个非常简洁的表达式. 令

$$L_n = \ln(p_n^0/p_n^1), \lambda_{mn} = \ln(q_{mn}^0/q_{mn}^1),$$

$$\lambda_n = \ln(q_n^0/q_n^1), \Lambda_{mn} = \ln(r_{mn}^0/r_{mn}^1).$$

对应的 BP 译码算法修改如下, 判决条件也相应的转换为对数域的形式 $LS(l)$:

1) 初始化

$$\lambda_{mn} = L_n, \Phi = \emptyset.$$

2) 计算每一比特的非法校验数目 $U(l)$

$$\mathbf{U}^T \cdot [(\mathbf{H} \cdot \mathbf{r})^T] \bmod 2 \cdot \mathbf{H} = \mathbf{Z}^T \cdot \mathbf{H}.$$

3) 校验点更新

$$\Lambda_{mn} = 2 \operatorname{ar} \tanh \left(\prod_{n' \in N(m) \setminus n \text{ 且 } n' \notin \Phi} \tanh(\lambda_{m'n'} / 2) \right).$$

4) 比特点更新

$$\lambda_{mn} = L_n + \sum_{m' \in M(n) \setminus m} \Lambda_{m'n}, \lambda_n = L_n + \sum_{m \in M(n)} \Lambda_{mn}.$$

5) 计算接收到的 l 的后验概率提供的可靠性信息 $LS(l)$, $LS(l) = |\lambda|$.

6) 记录下次迭代过程中不需要更新的比特 l . l 满足 $U(l) = 0, LS(l) > M$, 则把 l 加入集合 Φ . M 为

判断门限, $M > 0$).

7) 比特判决

若 $\lambda > 0$, x_n 判决为 0; 否则, x_n 判决为 1, $n = 1, 2, \dots, N$. 若 $H^T x = 0$ 或者迭代次数到达最大迭代次数, 则结束, x 作为译码输出; 否则转步骤 2) 继续迭代.

4 仿真结果

本节给出了试验仿真结果, 试验中采用码率 0.5 的规则 LDPC 码(4 096, 3, 6), 信道信噪比 $E_b/N_0 = 1.6$ dB, 每种译码算法中迭代次数都是 30 次. 图 3 和 4 分别给出了传统的 BP 译码算法和本文改进 BP 译码算法每次迭代过程中的计算量比较. 图中的纵坐标为归一化的计算量, 即设传统的 BP 译码算法的每次迭代中的计算量为 1. 从图中可明显地看出, 本文改进的 BP 译码算法, 随着迭代次数的增加, 每次迭代中的计算量明显下降.

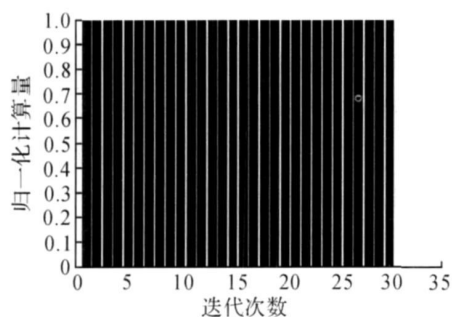


图 3 传统 BP 算法每次迭代的计算量

Fig. 3 Computational load in every iteration of traditional BP decoding

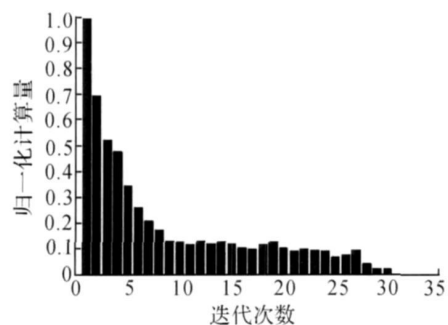


图 4 改进 BP 算法每次迭代的计算量

Fig. 4 Computational load in every iteration of proposed BP decoding

图 5 和 6 分别给出了传统的 BP 译码算法和本文改进的 BP 译码算法的译码效率图. 图中的纵坐标是每次纠正比特的个数与运算量的比值的归一化值, 即保证总的译码效率为 1. 可以看到传统的 BP 算法, 随着迭代次数的增加, 纠正的错误比特越来越

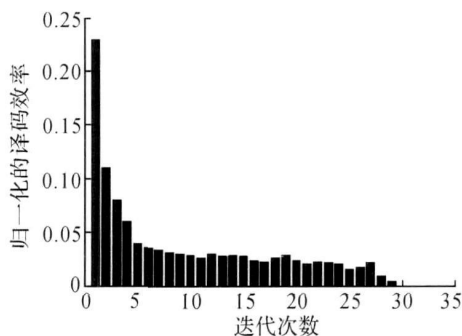


图 5 传统 BP 译码算法的译码效率

Fig. 5 Decoding efficiency of traditional BP decoding

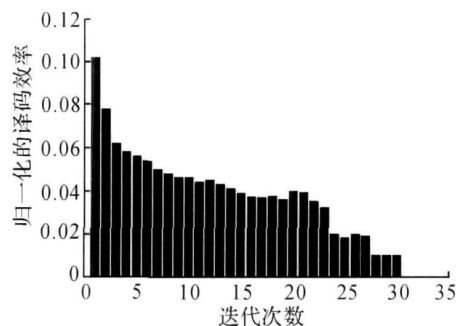


图 6 改进 BP 译码算法的译码效率

Fig. 6 Decoding efficiency of proposed BP decoding

少, 而每次迭代中的计算量完全一样, 因此, 译码效率随着迭代次数的增加而显著下降. 采用基于非法校验数和伪后验概率的改进 BP 算法后, 虽然随着迭代次数的增加纠正的比特数越来越少, 但是每次迭代的运算量也同样会随着迭代次数的增加而减少, 而两者比值的下降程度则明显放缓. 图 6 中的门限 $U(l) = 0, S(l) = 0.8$. 图 7 所示为门限取 $U(l) = 0, S(l) = 0.7$ 时的译码效率图, 可以看到当译码门限变松时, 译码效率可以进一步提高, 但效率提高是以译码性能下降为代价的.

图 8 给出了传统的 BP 译码算法和改进的 BP 译码算法的误码率 (BER) 性能比较. 可以看到本文改进的 BP 译码算法相对传统的 BP 译码算法, 在误

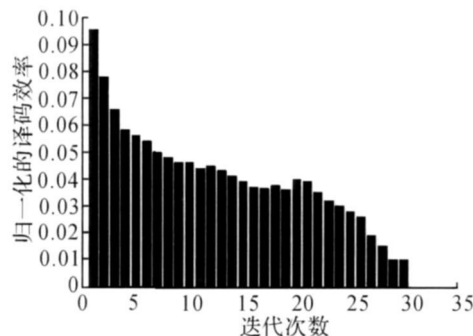


图 7 改进 BP 译码算法的译码效率

Fig. 7 Decoding efficiency of traditional BP decoding

码性能上的损失极小(当误码率 10^{-4} 时, 信噪比损失大约 0.4 dB), 而且随着信道条件的改善, 这种损失进一步减少. 这是因为, 随着信道环境的改善在每次迭代过程中伪后验概率的准确性加强. 因此, 误判的不更新的比特点数减少. 同时还可以看到随着门限的放松, 译码性能会逐渐下降, 这与实际中由于判别门限放松, 误判了一些本该更新的比特是有关.

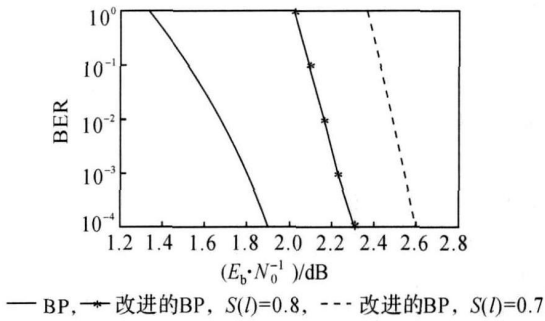


图 8 不同 BP 译码算法下误码率性能比较
Fig. 8 BER of different decoding algorithm

5 结 语

BP 译码算法效率不高的很大原因是在每一次迭代过程中都要对所有比特和校验信息进行更新. 本文提出了一种基于非法校验数和伪后验概率差的比特可靠性判别方法, 采用提出的方法来判断每个比特的可靠性. 在 BP 迭代译码过程中只需要更新那些有可能发生错误的比特的信息, 而无需更新可靠性非常高的比特的信息. 试验结果表明, 提出的方法大大地减少了迭代过程中的运算量, 提高了译码效率和收敛速度.

参考文献(References):

[1] GALLAGER R G. Low-density parity-check codes [J]. **IEEE Transactions on Information Theory**, 1961(1): 21-28.
[2] MACKAY D J C. Good error-correcting codes based on very sparse matrices [J]. **IEEE Transactions on Information Theory**, 1999, 3(45): 399-431.
[3] HEUM C C. Technical issues for enhanced and beyond

3G systems [C] // **The International Forum Future Mobile Telecommunications&China-EU Post Conference on Beyond 3G**. Beijing, China: [s. n.], 2002.
[4] FOSSORIER M P C, MIHALJEVIC M, IMAI H. Reduced complexity iterative decoding of low-density parity check codes based on belief propagation [J]. **IEEE Transactions on Communications**, 1999, 47(5): 673-680.
[5] CHEN J, FOSSORIER M P C. Near optimum universal belief propagation based decoding of low-density parity check codes [J]. **IEEE Transactions on Communications**, 2002, 3(50): 406-414.
[6] HU Xiao-yu, EYANGELOS E, DIETER-MICHAEL A, et al. Efficient implementations of the sum-product algorithm for decoding LDPC codes [C] // **GLOBECOM 2001 IEEE Global Telecommunications Conference**. [S. l.]: IEEE, 2001, 2: 1036-1036E.
[7] MAO Yong-yi, BANIHASHEMI A H. A new schedule for decoding low-density parity-check codes [C] // **GLOBECOM 2001 IEEE Global Telecommunications Conference**. [S. l.]: IEEE, 2001, 2: 1007-1010.
[8] ZHANG J, FOSSORIER M. Shuffled belief propagation decoding [J]. **IEEE Trans Communications**, 2005, 2(53): 209-213.
[9] LUBY M, MITZENMACHER M, SHOKROLLAHI A. Analysis of low density codes and improved designs using irregular graphs [C] // **Proc 30th Annu ACM Symp Theory of Computing** [S. l.]: ACM, 1998: 249-258.
[10] LUBY M, MITZENMACHER M, SHOKROLLAHI A. Improved low-density parity-check codes using irregular graphs and belief propagation [C] // **Proc 1998 IEEE Int Symp Information Theory**. Cambridge, MA: IEEE, 1998: 117.
[11] MACKAY D, WILSON S, DAYEY M. Comparison of constructions of irregular gallager codes [J]. **IEEE Trans Commun**, 1999, 10(47): 1449-1454.
[12] RICHARDSON T J, SHOKROKKAHI M A, URBANKE R L. Design of capacity approaching irregular low-density parity-check codes [J]. **IEEE Trans on Info Theory**, 2001(47): 619-637.