

用于 LDPC 码快速译码的改进多比特翻转算法

马克祥, 孙吉成, 王萌, 张雨明, 张海林

(西安电子科技大学 综合业务网理论与关键技术国家重点实验室, 陕西 西安 710071)

摘 要: 为了提高 RRWBF 算法的译码速度, 提出一种多比特翻转机制来加快 RRWBF 算法的收敛速度。但是, 使用该多比特翻转机制的 RRWBF 算法时, 译码过程中出现与单比特翻转类似的循环翻转现象, 影响其译码性能。为此, 进一步提出一种循环翻转消除机制来破坏多比特翻转译码过程中产生的循环翻转, 进而提高其译码性能。仿真结果显示, 与单比特翻转算法相比, 提出的基于循环翻转消除的多比特 RRWBF 算法以较小的译码性能损失换来译码速度的较大提升。

关键词: LDPC 码; 加权比特翻转算法; 快速译码

中图分类号: TN911.22

文献标识码: A

文章编号: 1000-436X(2014)02-0118-07

Improved multi-bits flipping algorithm for high-speed LDPC decoding

MA Ke-xiang, SUN Ji-cheng, WANG Meng, ZHANG Yu-ming, ZHANG Hai-lin

(State Key Lab. of Integrated Service Networks, Xidian University, Xi'an 710071, China)

Abstract: In order to improve the decoding speed of the RRWBF algorithm, a multiple-bits selection mechanism was proposed to accelerate the decoding convergence speed of the RRWBF algorithm. However, the RRWBF algorithm with the multiple-bits selection mechanism will appear the bit-repeated flipping, which can impair the decoding performance of the RRWBF algorithm. Hence, a bit-repeated flipping elimination mechanism was proposed to reduce the bit-repeated flipping in the process of the multiple-bits flipping decoding. Simulation results show that the RRWBF algorithm with the multiple-bits selection and the repeated flipping elimination mechanism has a much faster decoding speed at the cost of a little decoding performance loss compared with the RRWBF algorithm.

Key words: LDPC code; weighted bit flipping algorithm; high-speed decoding

1 引言

LDPC 码具有逼近香农限的良好译码性能, 在近年来得到了广泛的关注, 并且已经被纳入多种系统标准中, 比如 Wi-Fi(IEEE 802.11n)^[1]、10 Gbit Ethernet(IEEE 802.3an)^[2]、DVB-S2^[3]等。为了获取更好的译码性能, LDPC 码的各种译码算法得到了广泛研究, 其中最主要的 2 类是 belief propagation (BP)-based 算法和 bit flipping (BF)-based 算法。BP-based 算法是基于概率消息传递的软判决算法,

当码长较长时译码性能接近香农限, 但是运算量非常大, 必须使用 RAM 资源存储迭代过程中产生的大量的中间数据, 不利于工程实现^[4]; BF-based 算法是基于校验和迭代的硬判决译码方法, 算法操作简单易于工程实现, 可以在译码性能和实现复杂度之间进行有效折中。

由于在译码复杂度方面的固有优势, BF-based 算法得到广泛的研究。在 weighted BF (WBF)算法中, 通过给不同的校验方程引入不同的可靠性度量值进而有效地提高了 BF 算法的译码性能^[5]。但是

收稿日期: 2013-03-29; 修回日期: 2013-08-21

基金项目: 国家自然科学基金资助项目(61072069); 教育部科学技术研究重点(重大)基金资助项目(2010ZX03002-005); 高等学校学科创新引智计划基金资助项目(B08038)

Foundation Items: The National Natural Science Foundation of China (61072069); The National Science and Technology Major Project of the Ministry of Science and Technology of China (2010ZX03002-005); The 111 Project (B08038)

该算法并未对不同比特节点对校验失败影响的程度进行区分。理论上具有较小信道接收值的比特节点比具有较大信道接收值的节点对校验失败负更多的责任。根据上述特性，文献[6]提出一种新的品质因素函数去衡量比特的可靠性，然后翻转具有最大品质因素比特节点的硬判结果完成译码，该算法简记为 RRWBF。通过消除 RRWBF 算法品质函数中包含的冗余计算，C H Lee 提出一种实现有效的 RRWBF 算法^[7]，简记为 IRRWBF。在 IRRWBF 算法每次迭代过程中只翻转一个比特节点的硬判结果，进而在每次迭代中只有包含该比特的校验节点的校验和发生变化。因为在迭代过程中比特的品质因素随着校验结果的更新而变化，所以文献[8]只更新符号发生变化的品质因素分量完成所有比特品质因素的更新，从而大幅度降低品质因素更新模块的计算量。但是 IRRWBF 算法作为单比特翻转译码算法，其迭代译码的收敛速度较慢。

因此，越来越多的学者开始研究收敛速度快的多比特翻转译码算法。这些算法的主要区别就在于使用了不同翻转比特选择机制。文献[9,10]给出一种多比特翻转算法，该算法在选择下一个翻转比特时，需要利用码字的 Tanner 图生成当前翻转比特的低相关度集，再从集合中选择需要翻转的比特，其缺点是生成低相关度集的运算复杂，不利于硬件实现。文献[11]提供了 2 种多比特翻转算法，其中一种算法需要估计信道的噪声方差，故而其译码性能会受到噪声估计准确性的影响；另一种算法需要对候选集进行全排序，其实现时延和计算量较大，不适合于构造低功耗高速译码器。文献[12]中算法同时翻转品质因素值大于某一动态门限的比特完成快速译码，硬件实现复杂度相对较低，且译码性能及收敛速度也较为理想，因此本文将会与该算法进行详细的分析对比。为了提高 RRWBF 算法的译码速度，本文首先提出一种低复杂度的多比特翻转机制：利用校验式重量计算出每次迭代的翻转比特数目，找到相应数目的具有最大品质因素的比特同时翻转完成多比特翻转译码操作；另外，采用循环翻转消除机制来消除 RRWBF 算法的译码过程中出现的比特循环翻转的现象，实现了译码性能的提升。仿真结果显示，与文献[12]中算法相比，本文算法的译码收敛速度有明显的优势，更适用于构造低复杂度的高速 LDPC 译码器。

2 消除冗余计算的 IRRWBF 算法

本文主要研究二进制 (N, K) LDPC 码，对应的校验矩阵记为 $\mathbf{H}=[h_{MN}]$ 。编码后的码字 $\mathbf{c}=[c_1 \ c_2 \ \cdots \ c_N]$ 采用 BPSK 调制后，经 AWGN 信道传输得到的接收向量为 $\mathbf{y}=[y_1 \ y_2 \ \cdots \ y_N]$ 。 \mathbf{z}^k ($0 \leq k \leq K_{\max}$) 表示第 k 步迭代生成的硬判决向量，其中， K_{\max} 是译码的最大迭代次数。 \mathbf{g}_k 为伴随向量 $\mathbf{s}^k = \mathbf{z}^k \times \mathbf{H}^T$ 的重量。 $N(m) = \{n: h_{mn} = 1\}$ 表示与校验节点 m 相连的比特节点集合； $M(n) = \{m: h_{mn} = 1\}$ 表示与比特节点 n 相连的校验节点集合。 T_m 为校验节点 m 提供给与之相连的比特节点的可靠性度量值。比特节点 n 的品质因素 E_n 用来衡量该比特节点的可靠性，从而决定该比特节点的硬判结果在译码过程中是否需要翻转。 n_{fk} 表示第 k 次迭代时需要翻转的比特节点， $M(n_{fk})$ 为与之相连的 d_c 个校验节点组成的集合。相应地，在第 k 次迭代结束之后这些校验节点的校验结果也将会发生翻转，因此 $M(n_{fk})$ 被称为翻转校验集。对于 $M(n_{fk})$ 中的所有元素 $m_{f,k}$ ，与它们相连的比特节点组成的集合记为 $n_{m_{f,k}}$ 。这些比特节点在下次迭代过程中会发生品质因素的更新，因此称之为（品质因素）更新比特集 $N(m_{f,k})$ 。消除冗余计算的 RRWBF 算法详述如下。

初始化： $k=0$ ；计算每个比特节点 n 的初始硬判决值 $z_n^0 = (1 - \text{sgn}(y_n))/2$ 以及校验节点 m 提供的可靠性度量值， $T_m = \sum_{n \in N(m)} |y_n|$ 。计算初始译码序列的伴随向量， $\mathbf{s}^0 = \mathbf{z}^0 \times \mathbf{H}^T$ 。

step1 如果 $\mathbf{s}^k = \mathbf{0}$ ，停止译码并且输出 \mathbf{z}^k 。

step2 计算每个比特节点 n 的品质因素。

$$E_n^k = \begin{cases} \frac{1}{|y_n|} \sum_{m \in M(n)} (2s_m^k - 1)T_m, \\ (k = 0, n \in [0, N]) \\ E_n^{k-1} - 2 \frac{1}{|y_n|} \sum_{m \in M(n) \cap M(n_{fk})} (2s_m^{k-1} - 1)T_m, \\ (k > 0, n \in N(m_{f,k})) \end{cases} \quad (1)$$

step3 计算 $n^* = \arg \max E_n^k$ ，然后翻转比特节点 n^* 的译码结果。

step4 $k=k+1$ 。如果 $k > K_{\max}$ ，译码失败并且输出 \mathbf{z}^k ，否则计算校验节点在第 k 步迭代的伴随式

s^k , 计算公式为

$$s_m^k = \begin{cases} s_m^{k-1}, m \notin M(n_{f,k}) \\ 1 - s_m^{k-1}, m \in M(n_{f,k}) \end{cases} \quad (2)$$

然后转到 step1 进行下一次迭代译码尝试。

3 比特循环翻转现象分析

通过对 IRRWBF 算法译码失败帧进行仔细分析后发现, 造成 IRRWBF 算法译码失败的主要原因是出现了比特循环翻转现象, 即每次迭代都循环翻转相同的比特位置从而造成剩余的错误比特无法纠正直至译码迭代次数达到初始设定的最大值。循环的产生与码字结构以及硬判决向量密切相关, 详述如下。

3.1 单比特循环翻转现象

假设实际错误比特位置为 f_1 、 f_2 , 正确比特位置记为 t_1 。每次迭代只翻转一个比特位置。包含 f_1 、 f_2 和 t_1 的校验方程为

$$\begin{array}{l|ccc} & \mathbf{s}^k & \mathbf{s}_a^{k+1} & \mathbf{s}_b^{k+1} \\ \hline f_1 + t_1 + \dots = s_1 & 1 & 0 & 0 \\ f_2 + \dots = s_2 & 1 & 1 & 1 \\ t_1 + \dots = s_3 & 0 & 0 & 1 \\ f_1 + f_2 + \dots = s_4 & 0 & 1 & 0 \end{array}$$

第 k 次迭代校验式向量为 \mathbf{s}^k , 根据式(1), 可以将品质因数计算式化为

$$\begin{aligned} E_{f_1}^k &= [(2s_1^k - 1)T_1 + (2s_4^k - 1)T_4] / |y_{f_1}| \\ E_{f_2}^k &= [(2s_2^k - 1)T_2 + (2s_4^k - 1)T_4] / |y_{f_2}| \\ E_{t_1}^k &= [(2s_1^k - 1)T_1 + (2s_3^k - 1)T_3] / |y_{t_1}| \end{aligned}$$

为简化分析, 本文假设 $|y_{f_1}| = |y_{f_2}| = |y_{t_1}| = 1$, 仅研究 T_m 和 \mathbf{s}^k 在品质因数计算中的关系, 具体分析如下。

1) 比特正确翻转条件

在第 k 次迭代中, 若 f_1 的品质因数 $E_{f_1}^k$ 为三者之中最大, 则翻转比特位置 f_1 , 校验式将变为 \mathbf{s}_a^{k+1} ; 在第 $k+1$ 次迭代中, 若 f_2 的品质因数 $E_{f_2}^{k+1}$ 为最大值时, f_2 被翻转, 则译码成功。

针对上述假设条件, 对成功译码时 T_m 之间的关系推导如下: 第 k 次迭代中包含 2 个不等式 $E_{f_1}^k > E_{f_2}^k$ 、 $E_{f_1}^k > E_{t_1}^k$, 第 $k+1$ 次迭代中包含 2 个不等式 $E_{f_2}^{k+1} > E_{f_1}^{k+1}$ 、 $E_{f_2}^{k+1} > E_{t_1}^{k+1}$, 根据 s_i^k 和 s_i^{k+1} 的值, 品质因数计

算将进一步简化为

$$\begin{aligned} E_{f_1}^k &= T_1 - T_4 & E_{f_1}^{k+1} &= -T_1 + T_4 & T_1 - T_4 &> T_2 - T_4 \\ E_{f_2}^k &= T_2 - T_4 & \& E_{f_2}^{k+1} &= T_2 + T_4 & \Rightarrow T_1 - T_4 > T_1 - T_3 \\ E_{t_1}^k &= T_1 - T_3 & E_{t_1}^{k+1} &= -T_1 - T_3 & T_2 + T_4 &> -T_1 + T_4 \\ & & & & T_2 + T_4 &> -T_1 - T_3 \end{aligned}$$

根据 $T_m > 0$, 可以得到 $T_2 + T_4 > -T_1 + T_4 > -T_1 - T_3$, 进而可得 $T_1 > T_2$ 、 $T_3 > T_4$, 即接收向量实现成功译码的条件。由于 2 个不等式之间不存在矛盾关系, 显然存在 T_m 满足这 2 个不等式, 进而使得相应接收向量成功译码。

2) 比特循环翻转条件

出现循环翻转现象时, 相邻 2 次迭代连续翻转相同的比特位置。因此, 假设第 k 次迭代中品质因素 $E_{t_1}^k$ 最大, 即 $E_{t_1}^k > E_{f_1}^k$ 、 $E_{t_1}^k > E_{f_2}^k$, 翻转比特位置 t_1 , 这时相关校验向量将变化为 \mathbf{s}_b^{k+1} ; 在随后的第 $k+1$ 次迭代中, 假设品质因素 $E_{f_1}^{k+1}$ 仍最大, 即 $E_{f_1}^{k+1} > E_{f_2}^{k+1}$ 、 $E_{f_1}^{k+1} > E_{t_1}^{k+1}$, 翻转比特位置仍为 t_1 , 至此陷入循环翻转。

根据上述假设情况进行分析, 得到如下式子:

$$\begin{aligned} E_{f_1}^k &= T_1 - T_4 & E_{f_1}^{k+1} &= -T_1 - T_4 & T_1 - T_3 &> T_2 - T_4 \\ E_{f_2}^k &= T_2 - T_4 & \& E_{f_2}^{k+1} &= T_2 - T_4 & \Rightarrow T_1 - T_3 > T_1 - T_4 \\ E_{t_1}^k &= T_1 - T_3 & E_{t_1}^{k+1} &= -T_1 + T_3 & -T_1 + T_3 &> -T_1 - T_4 \\ & & & & -T_1 + T_3 &> T_2 - T_4 \end{aligned}$$

根据 $T_m > 0$, 整理上面不等式得到:

$$\begin{cases} T_3 < T_4 \\ T_2 < T_4 \\ T_1 - T_3 > T_2 - T_4 \\ T_3 - T_1 > T_2 - T_4 \end{cases}$$

这 4 个不等式相互间不存在矛盾, 例如当 $T_1 = 1.2T_2$ 、 $T_2 = T_2$ 、 $T_3 = 1.3T_2$ 、 $T_4 = 1.4T_2$ 时上述不等式成立。因此当 T_m 符合该条件后, 将会出现比特翻转循环。

3.2 循环翻转现象宏观分析

在第 k 次迭代中, 设翻转向量为 \mathbf{p}^k , 该向量的维度为 N 。 \mathbf{p}^k 中被翻转比特位置的值为 1, 其余比特位置值为 0, 由此可得 2 个等式

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \mathbf{p}^k \quad (3)$$

$$\mathbf{p}^k + \mathbf{p}^k = \mathbf{0} \quad (4)$$

循环比特翻转现象出现时: $\mathbf{p}^k = \mathbf{p}^{k+1}$, 进而 $\mathbf{z}^{k+2} = \mathbf{z}^{k+1} + \mathbf{p}^{k+1} = \mathbf{z}^k + \mathbf{p}^k + \mathbf{p}^{k+1} = \mathbf{z}^k$ 。由前式推导

可知译码过程进入比特循环翻转状态。根据上述推导过程，本文提出一种有效的循环翻转消除机制：当出现翻转循环现象后，改变下次迭代的翻转比特数，则 $p^k \neq p^{k+1}$ ，那么 $z^{k+2} \neq z^k$ ，进而可以有效消除出现的比特循环翻转。据此规律本文设计出了一种翻转比特数目自适应切换的 RRWBF (ASRRWBF, adapt switch RRWBF) 算法在加速算法的收敛速度的同时能够消除比特循环翻转现象。

4 ASRRWBF 算法

为了提高 RRWBF 算法的收敛速度，本文设计了一种适用于 RRWBF 算法的多比特翻转机制。此外，为提高多比特翻转译码的性能，提出了循环翻转消除机制。最后，依据多比特翻转机制和循环翻转消除机制，相应地提出一种翻转比特数目自适应切换的 RRWBF 算法，详述如下。

4.1 多比特翻转机制

利用 RRWBF 算法的品质因素函数衡量比特节点的可靠性时，品质因素值越大，相应比特节点的错误概率越高。故而，在本文算法中同时翻转品质因素最大的 p 个比特节点来实现 LDPC 码的快速译码。显然， p 取不同的值会对算法的译码性能造成不同影响。因此，必须选择合适的 p ，以期在保证译码性能的同时尽可能加速算法的收敛速度。实际译码过程中，错误比特的个数与校验方程伴随向量的重量正相关^[13,14]，由此可以利用伴随式的重量对错误比特数进行实时估计，因此可以利用伴随向量的重量计算每次迭代需要翻转的比特个数，即 $C \cdot \lfloor g_k / d_c \rfloor$ ， C 为正比系数。但是，仿真实验表明上述方法不完全适用于 RRWBF 算法。通过对译码过程进行研究发，在译码的初始阶段， g_k 较大，因此利用上述方法计算出来的翻转比特的数值较大，造成过多的错误翻转，进而导致译码性能下降。因此进一步引入翻转比特数上限 N ，限定每次迭代的翻转比特数不超过 N ，进而有效地减少译码初始阶段由翻转比特数过多而造成的错误翻转，从而达到提高算法译码性能的目的。

4.2 循环翻转消除机制

仿真发现，在多比特翻转译码过程中也会出现类似单比特的比特循环翻转现象。在下面例子中，以每次迭代同时翻转 2 个比特为例进行简要的说明。

$$\begin{array}{l}
 f_1 + \dots = s_1 \\
 f_2 + \dots = s_2 \\
 f_1 + f_2 + \dots = s_3 \\
 f_1 + t_1 + \dots = s_4 \\
 f_2 + t_1 + \dots = s_5 \\
 t_1 + \dots = s_6
 \end{array}
 \begin{array}{|c|c|}
 \hline
 \mathbf{S}^k & \mathbf{S}^{k+1} \\
 \hline
 1 & 0 \\
 1 & 1 \\
 0 & 1 \\
 1 & 1 \\
 1 & 0 \\
 0 & 1 \\
 \hline
 \end{array}$$

根据多比特翻转机制，在第 k 次迭代中，如果品质因数不等式为 $E_{f_1}^k > E_{f_2}^k$ 、 $E_{t_1}^k > E_{f_2}^k$ ，则翻转位置为 t_1 和 f_1 ，在第 $k+1$ 次迭代中，如果不等式满足 $E_{t_1}^{k+1} > E_{f_2}^{k+1}$ 、 $E_{f_1}^{k+1} > E_{f_2}^{k+1}$ ，则翻转位置仍为 t_1 和 f_1 ，进而出现循环翻转现象。类似单比特循环翻转分析方法，假设 $|y_{f_1}| = |y_{f_2}| = |y_{f_3}| = |y_{t_1}| = 1$ 。利用式(1)计算相应的品质因素值并代入以上 4 个不等式，化简可以得到以下不等式组：

$$\begin{array}{l}
 T_4 + T_6 > T_2 + T_3 \\
 T_4 + T_1 > T_2 + T_5 \\
 T_4 + T_5 > T_2 + T_1 \\
 T_4 + T_3 > T_2 + T_6
 \end{array}$$

4 个不等式之间无矛盾，例如当 $T_4 > T_2$ 而其他 T_m 值相等时上述不等式成立。即当 T_m 满足上述条件时，将会出现比特循环翻转。仿真实验表明，多比特翻转过程出现的循环翻转现象对译码性能造成的损害较大。因此，本文设计新颖的循环翻转消除机制，在保证多比特翻转算法快速译码特性的同时，尽可能提升其译码性能。

循环翻转消除机制的基本思路是：设置一个标志变量 $flag$ ，该值由连续 2 次迭代的比特翻转位置信息确定。当连续 2 次迭代的翻转比特位置不同时， $flag = 0$ 完全相同时， $flag = 1$ ；其作用是当 $flag = 1$ 时，进行单/多比特机制切换，当 $flag = 0$ 时，不进行切换。详细工作流程为：在译码的初始阶段采用多比特翻转，翻转比特数为 $f_0 = m_0$ (m_0 为根据上节多比特翻转机制计算出的翻转比特数)，并开始比较连续 2 次迭代的翻转比特位置。假设第 $k-1$ 次迭代时的翻转比特位置同第 $k-2$ 次相同，则令 $flag = 1$ 即切换为单比特翻转，此时第 k 次的翻转比特数 $f_k = 1$ ；而后仍继续比较翻转比特位置，当第 $k+t$ 次迭代前再次出现连续 2 次的翻转比特位置相同时，再次令

$flag=1$ ，切换为多比特翻转，此时翻转比特数 $f_{k+i} = m_{k+i}$ ，以此类推，直到译码成功或到达迭代上限为止。

4.3 ASRRWBF 算法流程

建议的 ASRRWBF 算法与 IRRWBF 算法具有相同的初始化操作，因此下面具体描述时不再给出。算法其他流程描述如下。

step1 如果若 $s^k = \mathbf{0}$ ，停止译码并且输出 \mathbf{z}^k 。

step2 对于每个变量节点 n ，按照式 (1) 计算其品质因素。

step3 进行单/多比特切换选择：根据标志变量 $flag$ 进行比特切换选择，若 $(f_{k-1} = 1) \oplus (flag = 1)$ 则翻转比特数 $f_k = 1$ ，否则翻转比特数 $f_k = m_k (m_k = C \lfloor g_k / d_c \rfloor; \text{若 } m_k > U \text{ 则 } m_k = N)$ 。找到具有 f_k 个具有最大品质因素的比特位置，将这些比特译码结果翻转完成本次译码迭代。

step4 将第 k 次翻转比特位置进行记录，并与第 $k-1$ 次翻转位置进行比较，若翻转位置完全相同，则令 $flag = 1$ ，否则 $flag = 0$ 。

step5 $k=k+1$ 。如果 $k > K_{\max}$ ，译码失败并且输出 \mathbf{z}^k ，否则计算校验节点在第 k 步迭代的伴随式 s^k ，计算公式为

$$s_m^k = \begin{cases} s_m^{k-1}, m \notin M(n_{f,k}) \\ 1 - s_m^{k-1}, m \in M(n_{f,k}) \end{cases}$$

然后转到 step1 进行下一次迭代译码尝试。式中 $M(n_{f,k})$ 是由翻转比特向量 $n_{f,k}$ 构成的翻转校验集。

5 仿真分析

本节对 IRRWBF 算法、文献[12]中提出的改进的 LDPC 码多比特翻转译码算法 (IMBF, improved multi-bits flipping algorithm)、ASRRWBF 算法以及不包括循环翻转消除机制的 ASRRWBF 算法 (以下简称 ASRRWBF-N) 进行仿真分析。仿真使用的码字分别为：1) Mackay 规则码^[15]，码长 816 bit，码率 1/2，行重 10，列重 5 (以下简称码 1)；2) PEG 规则码^[16]，码长 1 000 bit，码率 1/2，行重 10，列重 5 (以下简称码 2)；3) QC 准循环码^[16]，码长 1 008 bit，码率 1/2，行重 10，列重 5 (以下简称码 3)。测试数据经过 BPSK 调制后经由 AWGN 信道传输。译码器的最大迭代次数为 100。每个 SNR 点观察到 100 帧错误数据停止译码性能统计。ASRRWBF 算

法中 C 和 N 参数之间相互影响，本文采取二维遍历的方法确定两者的最优值。IMBF 算法^[12]中包含一个品质因素的正比系数，这里用参数 ν 表示，该参数采用遍历方法，以收敛速度或者译码性能作为标准进行选取。

5.1 码 1 译码性能分析

码 1 的仿真性能如图 1 所示，图中给出了 ASRRWBF 算法、选取相同参数的 ASRRWBF-N 算法、2 种不同参数的 IMBF 算法 ($\nu=0.2$ 时迭代次数较少， $\nu=0.8$ 时性能较好) 以及标准的 IRRWBF 算法的译码性能曲线。

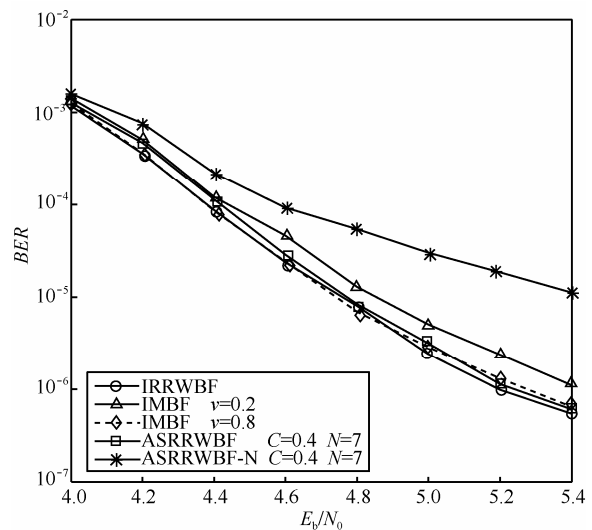


图 1 码 1 在不同算法下译码性能比较

1) 译码性能方面

在误码率为 10^{-4} 时，IMBF ($\nu=0.8$) 算法性能同 IRRWBF 算法基本一致，ASRRWBF 算法比 IRRWBF 算法性能下降约 0.02 dB；此时，IMBF ($\nu=0.2$) 算法比 IRRWBF 算法性能下降约 0.1 dB，而 ASRRWBF-N 的译码性能则要下降 0.2 dB。

在误码率为 10^{-5} 附近，ASRRWBF 算法、IMBF ($\nu=0.8$) 算法与 IRRWBF 算法三者性能差距较小，其中 IMBF ($\nu=0.8$) 算法性能略好，IMBF ($\nu=0.2$) 算法与 IRRWBF 算法的差距仍保持在 0.1 dB 左右，而未加入切换机制的单纯多比特译码算法 ASRRWBF-N 性能损失进一步增大。

在误码率为 10^{-6} 附近，可以清楚看到性能最优的仍为 IRRWBF 算法，其次为 ASRRWBF 算法，比 IRRWBF 算法差约 0.02 dB，再次是 IMBF ($\nu=0.8$) 算法，比 IRRWBF 算法差约 0.05 dB，IMBF ($\nu=0.2$) 算法则比 IRRWBF 算法差了 0.2 dB。

从整个信噪比区间上看 IRRWBF 算法具有最优的译码性能，ASRRWBF 算法的性能在低信噪比下略差于 IMBF($\nu=0.8$)算法，但在高信噪比下又略优于 IMBF($\nu=0.8$)算法，两者差距不大，IMBF($\nu=0.2$)算法较前面 3 种算法性能较差，而 ASRRWBF-N 算法的性能最差。

2) 迭代次数方面

根据表 1 数据可知，在信噪比 4.0 dB 附近，ASRRWBF 的迭代次数为 20.43 次，相比于 IRRWBF 的迭代次数 52.90 次，减少了 60%以上，相比于性能相近的 IMBF($\nu=0.8$)算法的迭代次数 39.05 次，也减少了近 50%，而 IMBF($\nu=0.2$)算法收敛速度较快，也仍需要 27.22 次。

在信噪比为 5.0 dB 时，ASRRWBF 的迭代次数仅为 11.88 次，相比于 IRRWBF 的迭代次数 31.88 次，仅为其 1/3，译码收敛速度有较大的提升。而与 ASRRWBF 算法性能相近的 IMBF($\nu=0.8$)算法需要的迭代次数为 22.80 次，是本文算法的 2 倍。收敛速度较快的 IMBF($\nu=0.2$)算法的迭代次数也需要 16.28 次，仍大于 ASRRWBF 算法。

综合来看，迭代次数最少的算法为 ASRRWBF 算法，其次为 ASRRWBF-N 算法，再次为 IMBF 算法，IRRWBF 算法迭代次数最大。

5.2 其他码字的仿真性能

通过对码 2、码 3 译码结果进行仿真发现：针对不同码字，IMBF 算法的最优性能时的系数 ν 不同，ASRRWBF 算法中，选取的 C 值无差别，但 N 值各不相同（如图 2、图 3 所示）。自适应切换算法对译码性能的提升与码 1 的情况一致。对于 ASRRWBF 算法，码 2 和码 3 性能方面最优的参数分别为 ($C=0.4, N=15$) 和 ($C=0.4, N=12$)。

对 3 种码字进行综合分析可知，ASRRWBF 算法在保持了多比特翻转机制速度优势的前提下，较好地提高了 LDPC 码的译码性能。该算法较

IRRWBF 性能下降约 0.03 dB，但同时算法所需的平均迭代次数下降约 2 至 3 倍，在有效地提高了 LDPC 码的译码速度的同时，尽可能减少译码性能的损失。

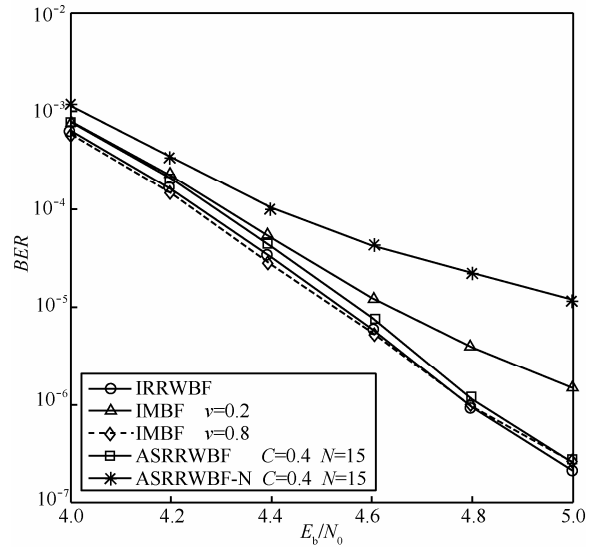


图 2 码 2 在不同算法下译码性能比较

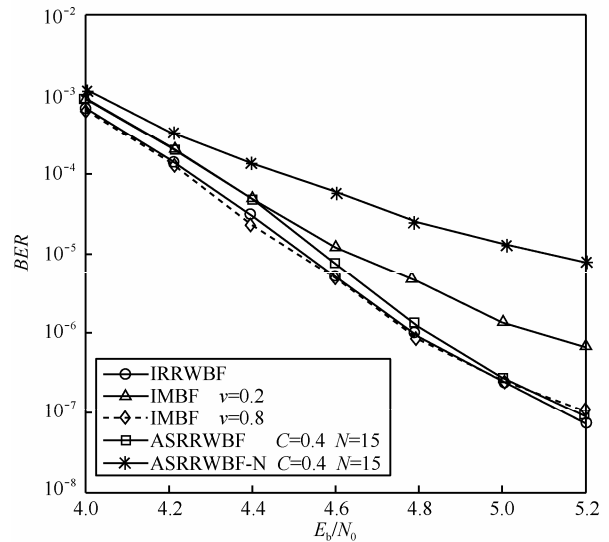


图 3 码 3 在不同算法下译码性能比较

表 1 码 1 在不同算法下平均迭代次数对比

信噪比	IRRWBF	ASRRWBF ($C=0.4, N=7$)	ASRRWBF-N ($C=0.4, N=7$)	IMBF ($\nu=0.2$)	IMBF ($\nu=0.8$)
4.0 dB	52.90	20.43	21.73	27.22	39.05
4.4 dB	42.77	15.23	15.67	20.48	30.34
4.8 dB	35.18	12.78	12.98	17.35	24.96
5.0 dB	31.88	11.88	12.01	16.28	22.80

6 结束语

本文提出一种多比特翻转机制来提高 IRRWBF 算法的收敛速度, 并且利用翻转比特数目的自适应切换来破坏 IRRWBF 算法译码过程中出现的循环翻转现象, 进而达到提高算法译码性能的目的。与 IRRWBF 算法相比, 仿真结果表明本文所提算法以较小的译码性能损失(约 0.03 dB)换得算法收敛速度的较大提高(约 2~3 倍), 可以更好满足光通信等高速数据传输通信系统快速译码的需求。

参考文献:

- [1] IEEE P802.11n/D2.00. Draft standard for Local and Metropolitan Area Networks, Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer(PHY) Specifications: Enhancements for Higher Throughput[S]. 2007.
- [2] IEEE Standard 802.3AN-2006. Specific requirements Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications Amendment 1: Physical Layer and Management Parameters for 10Gb/s Operation, Type 10GB ASE-T[S]. 2006.
- [3] EN 302 307 V1.1.1. Digital Video Broadcasting (DVB) Second Generation Framing Structure for Broadband Satellite Applications[S]. 2005.
- [4] CHEN X, KANG J, LIN S, *et al.* Memory system optimization for FPGA-based implementation of quasi-cyclic LDPC codes decoders[J]. IEEE Transactions on Circuits and Systems I: Regular Papers, 2011, 58(1):98-111.
- [5] KOU Y, LIN S, FOSSORIER M P C. Low-density parity-check codes based on finite geometries: a rediscovery and new results[J]. IEEE Transactions on Information Theory, 2001, 47(7):2711-2736.
- [6] GUO F, HANZO L. Reliability ratio based weighted bit-flipping decoding for low-density parity-check codes[J]. Electronic Letters, 2004, 40(21):1356-1358.
- [7] LEE C H, WOLF W. Implementation-efficient reliability ratio based weighted bit-flipping decoding for LDPC codes[J]. Electronic Letters, 2005, 41:1356-1358.
- [8] TIWARI H D, HUYNH N B, YONG B C. A parallel IRRWBF LDPC decoder based on stream-based processor[J]. IEEE Transactions on Parallel and Distributed Systems, 2012, 23(12):2198-2204.
- [9] HUNG J H, CHEN S G. A high-performance multibit-flipping algorithm for LDPC decoding[A]. Proceedings of the 2009 12th International Symposium on Integrated Circuits, ISIC'09[C]. Singapore, 2009. 151-154.
- [10] HUNG J H, CHEN S G. An effective multibit-flipping algorithm for LDPC decoding[A]. 2nd International Conference on Signal Processing and Communication Systems, ICSPCS 2008[C]. Gold Coast, QID, 2008.1-5.
- [11] HUANG Z, CHEN M, DIAO C, *et al.* Multiple candidates bit-flipping algorithms for decoding LDPC codes[EB/OL]. <http://onlinelibrary.wiley.com/doi/10.1002/ett.2605/pdf>, 2013.
- [12] 谢东觉, 张兴敢, 唐岚. 一种改进的 LDPC 码多比特翻转译码算法[J]. 现代电子技术, 2011, 34(3):11-13.

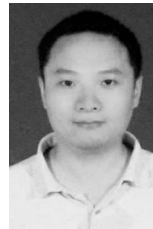
XIE D J, ZHANG X G, TANG L. An improved multi-bit flipping algorithm for LDPC decoding[J]. Modern Electronics Technique, 2011, 34(3):11-13.

- [13] CHO J, SUNG W Y. Adaptive threshold technique for bit-flipping decoding of low-density parity-check codes[J]. IEEE Communications Letters, 2010, 14(9): 857-859.
- [14] NGATCHED T M N, TAKAWIRA F, BOSSERT M. An improved decoding algorithm for finite-geometry LDPC codes[J]. IEEE Transactions on Communications, 2009, 57(2): 302-306.
- [15] MACKAY D J. Online database of low-density parity check codes[EB/OL]. <http://www.inference.phy.cam.ac.uk/mackay/codes/data.html>, 2013.
- [16] HU X, ELEFTHERIOU E, ARNOLD D M. Regular and irregular progressive edge-growth tanner graphs[J]. IEEE Trans Inf Theory, 2005, 51(1): 386-398.

作者简介:



马克祥(1986-), 男, 河南焦作人, 西安电子科技大学博士生, 主要研究方向为应用于无线通信系统和光通信系统的 LDPC 码字构造以及编译码算法。



孙吉成(1981-), 男, 江苏徐州人, 西安电子科技大学硕士生, 主要研究方向为 LDPC 码的低复杂度快速译码算法。



王萌(1990-), 男, 山东临沂人, 西安电子科技大学硕士生, 主要研究方向为 LDPC 码的快速译码算法以及 MIMO 检测技术。



张雨明(1988-), 男, 北京人, 西安电子科技大学硕士生, 主要研究方向为 LDPC 码的低错误平层译码算法。

张海林(1963-), 男, 陕西宝鸡人, 博士, 西安电子科技大学教授、博士生导师, 主要研究方向为宽带无线通信技术。