

DOI:10.19651/j.cnki.emt.2108087

基于FPGA的LDPC译码器的设计与实现

王兰珠 李锦明

(中北大学仪器与电子学院 太原 030000)

摘要: 为提高译码性能,基于CCSDS标准中应用于近地空间的(8 176, 7 154)LDPC码,根据归一化最小和译码算法理论,设计实现了尺度因子可变的LDPC译码器。本次译码器的设计主要对校验结点量化数据进行优化处理,设计实现了尺度因子随迭代次数变化而变化,且尺度因子值以2的倍数为基数,采用右移相加代替校验结点数据与尺度因子的乘法运算,简化硬件实现。此外,增加了译码校验模块来检验经校验结点与变量结点迭代计算后的码字是否译码成功,译码成功或到达设定的最大迭代次数后将数据发出。基于FPGA设计实现了LDPC译码器,其中硬件设计中采用部分并行的译码电路,合理利用硬件资源。在信噪比为1.8、最大迭代次数为15时,通过仿真及板级验证,并对比尺度因子值为0.5、0.75及尺度因子可变时的译码结果,证明了可变尺度因子NMS译码算法可以实现译码功能且具有较好的译码性能。

关键词: LDPC码;译码器;可变尺度因子;NMS译码算法;FPGA

中图分类号: TP2 **文献标识码:** A **国家标准学科分类代码:** 510.1050

Design and implementation of LDPC decoder based on FPGA

Wang Lanzhu Li Jinming

(School of Instrument and Electronics, North University of China, Taiyuan 030000, China)

Abstract: To improve the decoding performance, based on the (8 176, 7 154) LDPC code applied to near-earth space in CCSDS standard, and according to the normalized minimum sum (NMS) decoding algorithm, design and implement the LDPC decoder. The design of the decoder mainly optimized the quantization data of check nodes, the scale factor changes with the number of iterations, and the scale factor value is based on the multiple of 2 and used the right shift addition to replace the multiplication of check node data and scale factor, which simplifies the hardware implementation. In addition, add a decoding verification module to test whether the codeword is successfully decoded after iterative calculation of the check node and the variable node, and the data is sent out after successful decoding or reaching the set maximum number of iterations. The LDPC decoder is designed and implemented based on FPGA. In the hardware design, used parallel decoding circuits to make rational use of hardware resources. When the signal-to-noise ratio is 1.8 and the maximum number of iterations is 15, through simulation and board-level verification, and comparing the decoding results when the scale factor value is 0.5, 0.75 and the scale factor is variable, it is proved that the variable scale factor NMS decoding algorithm can realize the decoding function and has good decoding performance.

Keywords: LDPC code;decoder;variable scale factor;NMS decoding algorithm;FPGA

0 引言

低密度奇偶校验码(low density parity check code, LDPC),由于其具有接近香农极限的纠错性能^[1],在通信领域有良好的应用前景,获得了众多通信标准组织的认可,目前已被应用于DVB-S2^[2-3]、IEEE802.16e^[4]、IEEE802.11^[5]等通信标准。另外LDPC码也凭借自身的优异性能在卫星通信领域大放异彩^[6-9],2011年,我国在研究嫦娥二号通信

方案时采用LDPC码;2016年LDPC码首次打败广泛运用于三、四代移动通信的Turbo码,成为第五代移动通信的信道编码方案之一^[10-11],且成为主流空间无线通信系统中的关键技术。LDPC在航天和5G领域的亮眼表现奠定了后续关于信道编码研究中的地位^[12-15]。虽然LDPC码具有很好的抗干扰性能,但在提升译码性能的基础上降低译码复杂度是其硬件研究的重难点。

传统的归一化最小和(normalized minimum sum,

NMS)译码算法研究中,其尺度因子大多取定值、利用高精度浮点数进行计算,但高精度浮点数在硬件上很难实现。文献[16]采用整数运算的最小和译码算法,将译码算法中的变量用整数表示,然后量化为二进制进行译码运算;文献[17]提出 NMS 算法中尺度因子取值为 0.8 时具有更好的译码纠错性能;文献[18]对分层 NMS 算法进行了 FPGA 实现,其中尺度因子值为 0.75;文献[19]讲述了 Viterbi 译码原理,并采用 FPGA 内部寄存器资源并行处理的译码方法。随着 FPGA 器件应用的日渐增加^[20],为了提高译码性能,简化硬件实现,本文在对国际空间数据系统咨询委员会(Consultative Committee for Space Data Systems, CCSDS)标准中(8 176, 7 154) LDPC 码 NMS 译码算法的 FPGA 实现中,对校验点数据进行优化处理,将尺度因子以 2 的倍数为基数来取值,且其值随迭代次数变化而变化,并对量化后数据采用移位相加来处理校验结点更新运算,使得本文设计的 LDPC 码译码器在不增加硬件实现复杂度的基础上具有更好的译码性能。

1 LDPC 码结构与译码算法

1.1 LDPC 码结构

本文研究对象为(8 176, 7 154) LDPC 码,在 7 154 位原始信息码的基础上经编码后生成 1 022 位校验码,共 8 176 位数据,通过信道传输后进行迭代译码。其中校验矩阵 H 的构造对编译码过程起到了决定性作用,本次研究的 LDPC 码其 H 矩阵大小为 $1\ 022 \times 8\ 176$, 矩阵元素均由“0”、“1”构成,且首行非零元素位置已知,其余各行元素为首行数据循环右移得到,为方便研究,将该矩阵划分为 32 个 511×511 的循环子矩阵,每个循环子矩阵的位置及其首行非零元素位置如表 1 所示。

表 1 每个循环子矩阵的位置及其首行非零元素位置

循环子矩阵的位置及其首行非零元素位置			
$A_{1,1}(0,176)$	$A_{1,2}(12,239)$	$A_{1,3}(0,352)$	$A_{1,4}(24,431)$
$A_{1,5}(0,392)$	$A_{1,6}(151,409)$	$A_{1,7}(0,351)$	$A_{1,8}(9,359)$
$A_{1,9}(0,307)$	$A_{1,10}(202,457)$	$A_{1,11}(0,247)$	$A_{1,12}(18,281)$
$A_{1,13}(0,399)$	$A_{1,14}(202,457)$	$A_{1,15}(0,247)$	$A_{1,16}(36,261)$
$A_{2,1}(99,471)$	$A_{2,2}(130,473)$	$A_{2,3}(198,435)$	$A_{2,4}(260,478)$
$A_{2,5}(215,420)$	$A_{2,6}(282,481)$	$A_{2,7}(48,396)$	$A_{2,8}(193,445)$
$A_{2,9}(273,430)$	$A_{2,10}(302,451)$	$A_{2,11}(96,379)$	$A_{2,12}(191,386)$
$A_{2,13}(244,467)$	$A_{2,14}(364,470)$	$A_{2,15}(51,382)$	$A_{2,16}(192,414)$

对于任意一个循环子矩阵,其行重列重均为 2,即每行有两个非零元素、每列有两个非零元素,则整个校验矩阵 H 行重为 32、列重为 4,故本文研究的 LDPC 码是一种不规则 LDPC 码。

1.2 NMS 译码算法

NMS 译码算法是一种软判决译码算法,码长越长越接

近香农极限,且其算法简单,易于硬件实现。假设要进行译码的信息序列为 $x=(x_1, x_2, x_3, \dots, x_{8176})$, 经过核心算法校验结点与变量结点的迭代计算后,译码器译码输出的码字序列为,下面简要介绍该译码算法原理,算法中需要用到的变量含义如下。

$L(P_i)$:待译码的信息序列;

$L^{(l)}(r_{ij})$:第 l 次迭代时,变量结点传向校验结点的消息;

$L^{(l)}(q_{ij})$:第 l 次迭代时,校验结点传向变量结点的消息;

$L^{(l)}(q_i)$:对所有变量结点计算 l 次迭代后的硬判决消息;

$C(i)$:与变量结点 i 相连的校验结点的集合;

$C(i) \setminus j$:除校验结点 j 外与变量结点 i 相连的校验结点的集合;

$R(j)$:与校验结点 j 相连的校验结点的集合;

$R(j) \setminus i$:除变量结点 i 外与校验结点 j 相连的校验结点的集合;

则译码步骤为:

1)初始化,即计算变量结点接收到的信道消息值

$$L^{(0)}(q_{ij}) = L(P_i) = x \quad (1)$$

2)利用变量结点按行更新校验结点信息

$$L^{(l)}(r_{ij}) = \prod_{i' \in R(j) \setminus i} \text{sgn}(L^{(l-1)}(q_{ij})) * \min_{i' \in R(j) \setminus i} (|L^{(l-1)}(q_{ij})|) * \alpha \quad (2)$$

其中,为增强译码器译码性能,且在不增加硬件实现难度的基础上,尺度因子基于 2 的倍数取值,采用右移相加的算法简化校验结点处理计算,根据文献[21],本次尺度因子 α 值结果如下:

$$\alpha = \begin{cases} \frac{1}{2}, & n = 1 \\ \frac{1}{2} + \frac{1}{4} + \frac{1}{2^{n+1}}, & n = 2, 3, 4, 5 \\ \frac{1}{2} + \frac{1}{4}, & n > 5 \end{cases} \quad (3)$$

3)利用校验结点按列更新变量结点信息

$$L^{(l)}(q_{ij}) = L(P_i) + \sum_{j \in C_i \setminus j} L^{(l)}(r_{ji}) \quad (4)$$

4)码字判决

$$L^{(l)}(q_i) = L(P_i) + \sum_{j \in C_i} L^{(l)}(r_{ji}) \quad (5)$$

若 $L^{(l)}(q_{ij}) > 0$, 则 $\hat{C}_i = 0$, 否则 $\hat{C}_i = 1$ 。

5)译码校验

若译码后码字序列 \hat{C} 和检验矩阵 H 满足 $H * \hat{C}^T = 0$, 则译码成功,否则进行下一次迭代计算,当达到设定的最大迭代次数,译码失败。

2 LDPC 译码器的硬件设计与实现

本文用 ZYNQ7020 开发板实现(8 176, 7 154) LDPC

译码器的设计,利用 H 矩阵的稀疏特性,设计了数据存储模块、校验结点信息处理模块、变量结点信息处理与码字判

决模块、校验模块和控制模块。为便于测试译码器的功能,还设计了串口收发模块,译码器整体框图如图 1 所示。

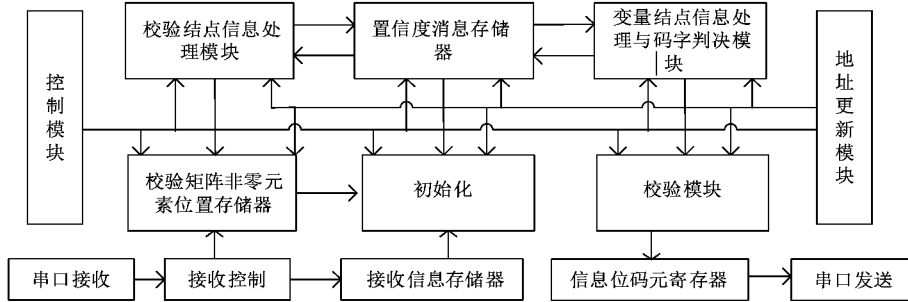


图 1 译码器整体框图

由于编码后数据在传输过程中,原来“0”、“1”的数据由于受到各种噪声干扰而变为取值在 $(-4,4)$ 的浮点型数据,为便于硬件实现,编码后的 8 176 位数据每一位在接收前均经上位机量化为 8 位二进制,其中包括 1 位符号位、3 位整数位、4 位小数位。首先 FPGA 通过串口接收待译码的 $8\ 176 \times 8$ 位数据,然后结合校验矩阵非零元素位置进行置信度消息初始化,之后按行进行校验结点信息处理,得到更新后的置信度消息,再进行变量结点信息的处理,一方面更新置信度消息,一方面进行码字判决,若校验成功,则译码成功,将译码后的信息序列通过串口发出,若校验失败,则继续进行校验结点与变量结点的迭代处理,直至达到设置的最大迭代次数,译码失败并通串口输出最终译码结果。

2.1 数据存储与初始化

本次设计的存储模块共分为 3 部分,包括接收数据的存储、校验矩阵非零元素位置的存储以及置信度消息矩阵的存储。接收数据共 8 176 个数据,每个数据由 8 位二进制组成,为方便并行读取数据,经接收控制分别存入 16 个位宽为 8、深度为 511 的双端口 RAM 中,其存储结构如图 2 所示。

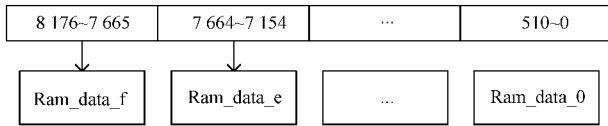


图 2 接收数据存储

校验矩阵是一种稀疏矩阵,每个子矩阵每行每列有两个非零元素,且子矩阵也遵循首行循环右移构成整个子矩阵,对每个子矩阵非零位置的存储分别用两个位宽为 9、深度为 511 的双端口 ROM 存储,第 1 个子矩阵中非零元素位置分布如图 3 所示,其在 ROM 中的位置存储如图 4 所示,校验矩阵的 32 个子矩阵共需 64 个 ROM 存储其位置信息,分块后第 1 行子矩阵信息分别存入 Hrom1_0_1~Hrom1_f_2;分块后第 2 行子矩阵信息分别存入 Hrom2_0_1~Hrom2_f_2。

置信度消息矩阵的大小及其非零元素个数和位置均

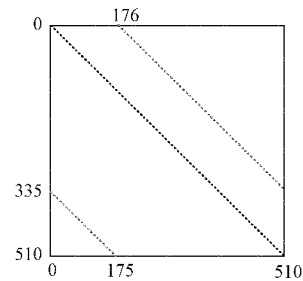


图 3 第 1 个子矩阵非零元素位置分布

Hrom1_0_1	Hrom1_0_2
0	176
1	177
⋮	⋮
334	510
335	0
⋮	⋮
509	174
510	175

图 4 第 1 个子矩阵非零元素位置存储

与校验矩阵相同,故本次设计存储结构与校验矩阵结构相同,共需要 64 个位宽为 8、深度为 511 的双端口 RAM 存储置信度消息矩阵,通过对校验结点与变量结点的迭代计算来更新置信度消息。

首次置信度消息矩阵是通过初始化单元得来的。待译码数据个数与校验矩阵每一行元素个数相等,首先根据校验矩阵每一行非零元素列位置,找到对应的待译码数据,存入对应位置的置信度消息矩阵中,即将校验矩阵每行中的非零元素替换为对应的待译码数据,即可得到初始化后的置信度消息矩阵。

2.2 校验结点信息处理模块

校验结点信息处理是对置信度消息矩阵的横向处理,参与运算的是矩阵中每行 32 个非零数据,通过上节存储模块的特点易设计校验结点信息处理模块的并行度位 2,每个并行处理模块完成同一存储地址的 32 个置信度消息

存储器数据的更新。校验结点信息处理电路设计如图5所示,其中CNPdatain10~CNPdatain1f是第1组32个输入数据,CNPdatain20~CNPdatain2f是第2组32个输入数据,每组输入数据端口16根线,每根线的高八位、低八位分别表示一个输入数据,与这64个输入数据相连的是64个置信度存储器的读数据端口;同理CNPdataout10~CNPdataout2f是与64个置信度存储器的写数据端相连的输出数据端口,用于将模块处理完成后的数据输入存储器更新原地址数据;mode即参与运算的尺度因子选项,mode随迭代次数的变化而变化,其每次迭代尺度因子取值如式(3)所示。

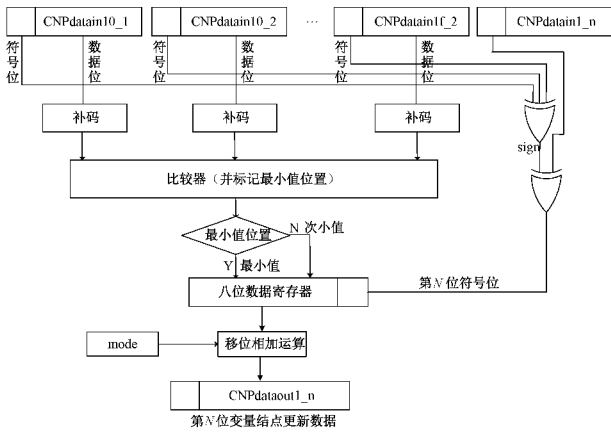


图5 校验结点信息处理电路设计

每个运算模块首先计算出32位数据的符号乘积sign,更新后的每一位数据符号即为sign与当前符号位乘积;其次需找出32位数据中的最小值和次小值,并标记出最小值位置,则更新后在非最小值位置的值为更新后符号与最小值乘积,在最小值位置的值为更新后符号与次小值乘积。在本次设计中,0、1的乘法运算可用异或运算代替,既不影响运算结果,也可以减少资源的使用,两组运算同时进行,运算结束后即把置信度消息矩阵更新成功。

2.3 变量结点信息处理与码字判决模块

变量点信息处理模块是对置信度消息矩阵的纵向处理,模块电路设计如图6所示,参与运算的是置信度消息矩阵每列的4个非零数据,由于码字判决参与运算的数据与变量结点信息处理基本相同,因此将两个模块放在一起设计,根据其存储特点,变量结点信息处理与码字判决模块的并行度设计为16,每个并行处理模块完成4个置信度消息存储器数据的更新并生成511位判决后的码字信息,其中fdata表示从接收数据存储器获得的初始置信度信息,check_out表示每个并行模块中通过码字判决产生的码字信息位。

对于每列非零元素的读取,根据每个511×511的置信度消息存储器,其首行两个非零元素的位置固定,本文采用如式(6)所示方法由列地址求出行地址:设校验矩阵的一个子矩阵首行非零元素列位置为n₁、n₂,则第n列两个

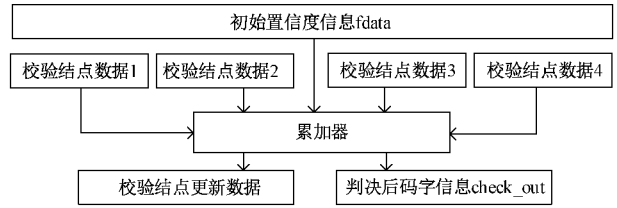


图6 变量结点信息处理与译码判决模块电路设计图

非零元素行地址m₁、m₂的值如式(6)所示。如第1个校验矩阵子矩阵首行非零元素列位置为0、176,则第0列对应非零元素行地址m₁ = 511 - 0 + 0 - 511 = 0, m₂ = 511 - 176 + 0 = 335,故可得出其他子矩阵的任意列对应的非零元素行地址。

$$\begin{aligned}
 m_1 &= \begin{cases} 511 - n_1 + n - 511, & 511 - n_1 \geq 511 \\ 511 - n_1 + n, & 511 - n_1 < 511 \end{cases} \\
 m_2 &= \begin{cases} 511 - n_2 + n - 511, & 511 - n_2 \geq 511 \\ 511 - n_2 + n, & 511 - n_2 < 511 \end{cases}
 \end{aligned} \tag{6}$$

2.4 译码校验模块

经码字判决得到的码字序列是否译码正确,需要进一步进行校验,通过移位寄存器得到判决后的码字序列分别存储在dataout0~dataoutf寄存器中,将其分别与校验矩阵的行向量相乘,由于检验矩阵元素的0、1特点,该矩阵乘法可简化为找到校验矩阵每行非零元素位置对应的码字并对其求异或运算,若求得所有结果均为0,则译码成功,否则译码失败,重新进行校验结点信息处理进行迭代译码,直至达到最大迭代次数。当检测到译码成功信号,截取前7154位信息码元通过串口发送出去。

2.5 控制模块

控制模块是整个系统的核心,控制着各个子模块的运行,图7为控译码控制流程。当接收完待编码信息后,根据接收的待编码信息进行置信度矩阵的初始化;初始化结束后,进行第1次迭代处理,其中迭代计算包括校验结点信息的处理、变量结点信息处理及码字判决,此时校验结点信息处理中的尺度因子值为;第1次校验结点与变量结点的迭代计算结束后,进行译码校验,若校验无误,则译码成功,将译码结果通过串口发送至上位机,否则进行下一次校验结点与变量结点的迭代计算,直至达到设定的最大迭代次数。根据本文实现的可变尺度因子NMS译码算法,在处理校验结点信息时,尺度因子的值随迭代次数的变化而变化,第1次迭代尺度因子值为1/2,第2~5次迭代尺度因子根据公式(1/2 + 1/4 + 1/2ⁿ⁺¹)而改变,其中n为迭代次数,当迭代次数大于5时,尺度因子取定值(1/2 + 1/4)。

3 测试与验证

本节主要完成译码器的功能测试,并进行结果分析及

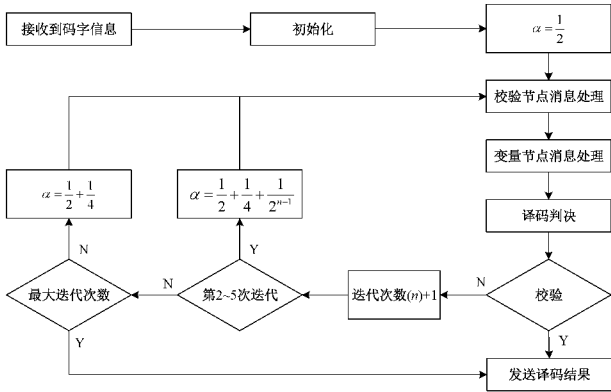


图 7 译码控制流程

功能验证。编码后的 8 176 位数据,通过二进制相移键控 (binary phase shift keying, BPSK) 调制,即完成 $0 \rightarrow 1$ 、 $1 \rightarrow -1$ 的元素映射后,为这组数据添加一组零均值、单位方差的正太分布随机数,即模拟加性高斯白噪声信道传输,接收端完成 BPSK 解调并将每一位数据量化为 8 位二

进制后通过串口送入译码器进行迭代译码,本次测试系统框图如图 8 所示。

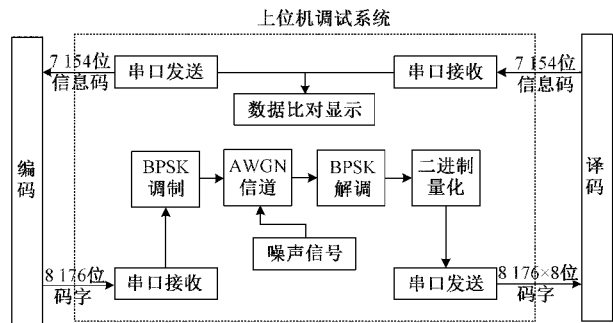


图 8 LDPC 译码测试系统框图

本次仿真主要测试了译码器的功能,图 9 为译码器仿真图,易看出尺度因子随迭代次数变化而变化,迭代 11 次后译码成功,译码无误,其中 code 为译码成功后的码字序列,与编码前码字序列相对比,译码无误。

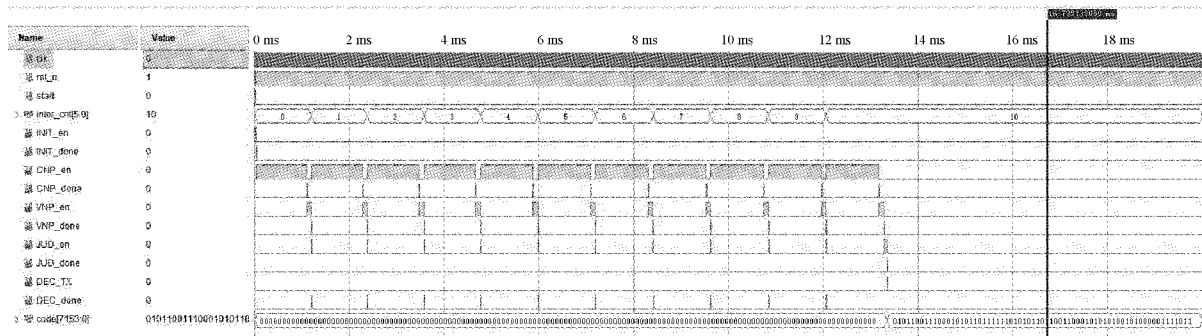


图 9 译码器仿真图

为实现译码器的板级验证,将译码器程序下载入 FPGA (zynq7020) 开发板,然后将编码加噪量化后的 65 408 位待译码数据通过串口助手发送至开发板进行译码,并接收译码完成数据,图 10 为译码器串口数据发送与接收显示图。将串口接收到的译码后码字保存在 txt 文本

文件中,随后导入 MATLAB 与编码前数据进行对比,两组数据完全一致,说明译码成功。



图 10 串口显示

为进一步验证译码性能,在信噪比为 1.8 dB、最大迭代次数设为 15 的情况下,通过对比尺度因子值分别为 0.5、0.75 与本文提出的可变尺度因子 3 种算法,测得迭代次数与误码个数的关系如图 11 所示,可以看出,随着迭代次数的增加,误码个数在降低,直至第 7 次误码个数降为 0,成功译码,相比于尺度因子值为 0.5 时减少两次迭代,相比于尺度因子值为 0.75 时,其误码个数降低的速率明显

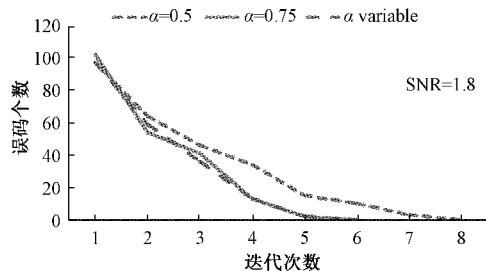


图 11 迭代次数与误码个数关系

加快。故不论从迭代次数还是从误码数降低的速率来看,本次设计的译码器均具有较好的译码性能。

4 结 论

本文基于可变尺度因子 NMS 译码算法对 CCSDS 标准下用于近地空间通信的(8 176, 7 154)LDPC 码进行译码器的 FPGA 设计实现,其中对校验结点的处理进行了优化,使尺度因子随迭代次数变化而变化,并用右移相加来代替校验结点处理时尺度因子的乘法运算,简化了硬件处理,基于 FPGA 设计实现了 LDPC 译码器。此译码器核心是校验结点信息与变量结点信息的迭代处理,其中校验点信息处理模块并行度为 2,变量点信息处理模块并行度为 16,同时设计了校验模块,检测到译码成功后即将译码成功的码字序列通过串口发送出来,否则达到设定的最大迭代次数,发送最后一次译码数据。通过仿真及板级测试,本次设计的译码器在实现译码功能的基础上具有较好的译码性能。

参考文献

- [1] MACKAY D J C, NEAL R M. Near Shannon limit performance of low density parity check codes [J]. *Electronics letters*, 1996, 32(18): 1645.
- [2] 刘盟,李家强,李杰,等. 基于 DVB-S2 协议的 LDPC 码译码算法研究 [J]. *微电子学与计算机*, 2019, 36(3): 71-76.
- [3] MARCHAND C, BOUTILLON E. LDPC decoder architecture for DVB-S2 and DVB-S2X standards [C]. 2015 IEEE Workshop on Signal Processing Systems (SiPS), IEEE, 2015: 1-5.
- [4] LU Q, SHAM C W, LAU F C M. Rapid prototyping of multi-mode QC-LDPC decoder for 802.11 n/ac standard [C]. 2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC), IEEE, 2016: 19-20.
- [5] 宁平. IEEE 802.16e 标准中 LDPC 编码的实现与仿真 [J]. *电子技术应用*, 2014, 40(9): 101-104.
- [6] 吴文懿. 低轨卫星通信 LDPC 码编译码算法研究 [D]. 哈尔滨: 哈尔滨工业大学, 2017.
- [7] 韩超. 面向 CCSDS 标准的深空通信信道编译码算法研究 [D]. 哈尔滨: 哈尔滨工业大学, 2013.
- [8] 刘超波. 深空通信中 LDPC 码构造和译码关键技术研究 [D]. 长沙: 湖南大学, 2012.
- [9] 侯毅, 刘荣科, 彭皓, 等. 适用于空间通信的 LDPC 码 GPU 高速译码架构 [J]. *航空学报*, 2017, 38(1): 236-245.
- [10] 张长青. 面向 5G 的 LDPC 码正则校验矩阵设计研究 [J]. *邮电设计技术*, 2020(1): 38-44.
- [11] 孟嘉慧, 赵旦峰, 田海. 面向 5G 的多元 LDPC 改进译码算法的仿真研究 [J]. *计算机科学*, 2018, 45(9): 141-145.
- [12] 徐恒舟, 李楠, 赵可新, 等. 5G 通信中准循环 LDPC 码的环结构分析 [J]. *电子测量与仪器学报*, 2019, 33(7): 50-55.
- [13] 穆锡金. 面向 5G 通信系统的信道编码技术研究 [D]. 西安: 西安电子科技大学, 2017.
- [14] 东梦楠. LDPC 码的编译码及其在 5G 系统中的应用研究 [D]. 北京: 北京邮电大学, 2018.
- [15] 葛广君, 殷柳国. 卫星高速数传系统多码率融合 LDPC 编码器设计 [J]. *清华大学学报(自然科学版)*, 2016, 56(6): 656-660.
- [16] 陈正康, 张会生, 李立欣, 等. LDPC 码最小和译码算法的整数量化 [J]. *系统工程与电子技术*, 2015, 37(10): 2371-2375.
- [17] 王昊. 5G 终端模拟器-LDPC 译码的研究与实现 [D]. 重庆: 重庆邮电大学, 2020.
- [18] 曾海粮. 一种多码率 LDPC 编译码关键技术研究 [D]. 成都: 电子科技大学, 2020.
- [19] 牛毅, 马忠松. 基于 FPGA 的高速 Viterbi 译码器 [J]. *国外电子测量技术*, 2011, 30(8): 63-65.
- [20] 廖永波, 李平, 阮爱武, 等. FPGA 中宽边译码器的测试方法研究 [J]. *仪器仪表学报*, 2010, 31(7): 1638-1643.
- [21] 李锦明, 王国栋, 刘梦欣, 等. CCSDS 标准下 LDPC 码的编译码算法研究 [J]. *电子学报*, 2020, 48(11): 2114-2121.

作者简介

王兰珠, 研究生, 主要研究方向为通信信号处理和 IC 设计。
E-mail: 994106986@qq.com

李锦明, 博士后, 副教授, 硕士生导师, 主要研究方向为动态测试、智能仪器技术。
E-mail: lijnming@nuc.edu.cn