

DOI:10.19651/j.cnki.emt.2105733

# 基于循环神经网络的低复杂度最小和译码算法

孙浩杰 夏巧桥

(华中师范大学 武汉 430079)

**摘要:** 针对神经网络信念传播译码算法具有高复杂度的问题,提出一种新的低复杂度的深度学习译码算法,降低译码方法硬件实现时的复杂度。通过给出置信传播译码算法替代图形,结合 Min-sum 算法,去除了双曲函数运算并将工程实际中需要消耗大量资源的乘法操作转换为简单的加法操作。结合循环神经网络结构,将多层不同的参数约束成单层的参数。结合不同信息的属性,不对来自信道的对数似然比消息附加额外的参数,只是将参数附加于校验节点更新时的边缘上。结合边缘参数的分布,发现部分权重数值与 1 偏离很大,判定为需要添加权重的边缘。提取有效深度神经网络权重进行训练来降低译码网络的参数数目。经实验验证,所提出的译码方法有效地降低了神经网络的译码复杂度,将神经网络参数减少约 20%。并且算法与置信传播译码相比,在高信噪比区域取得 1 dB 的性能提升,便于硬件实现,具有较强的实用性。

**关键词:** 深度学习;信道译码;Tensorflow

**中图分类号:** TN911.22 **文献标识码:** A **国家标准学科分类代码:** 510.5025

## Low complexity minimum sum decoding algorithm based on recurrent neural network

Sun Haojie Xia Qiaoqiao

(Central China Normal University, Wuhan 430079, China)

**Abstract:** A new low-complexity deep learning method is proposed to improve the complex neural network belief propagation algorithm, and reduce the complexity of the hardware implementation. By giving an alternative graph of the confidence propagation decoding algorithm and combining with the Min-sum algorithm, the hyperbolic function operation is removed and the multiplication operation which consumes a lot of resources in engineering practice is converted into a simple addition operation. And combined with the recurrent neural network, different parameters of multiple layers are constrained into single layer parameters. Considering the attributes of different information, no additional parameters are attached to the log-likelihood ratio message from the channel, parameters are attached to the edge when the check node is updated. Combined with the distribution of edge parameters, it is found that part of weight values deviate greatly from 1, which is determined as the edge that needs to add weight, parameters of neural network are reduced by about 20%. performance of the proposed algorithm is improved by 1 dB compared with the belief propagation algorithm in the high SNR regime, which is convenient for hardware implementation and has strong practicability.

**Keywords:** deep learning; channel decoding; Tensorflow

## 0 引言

随着 5G 通信系统<sup>[1]</sup>的提出,通信业务对高传输速率、低时延、低功耗和高可靠性的传输需求越来越迫切。现有的信道译码算法逐渐不能满足要求。

LDPC 译码<sup>[2]</sup>由于可以在信宿端检测并且纠正信号<sup>[3]</sup>在传输过程中产生的误码,能够保证数据传输的可靠性,具

有优异的性能。因此成为学术界及工业界研究的重点。

在过去的几年中,深度学习已在推理和建模任务中实现了巨大的进步,例如语音识别,图像中的目标识别。而将深度学习应用于 LDPC 码的置信传播(BP)解码器的设计,已证明前景广阔。Nachmani 等<sup>[4]</sup>表明如果将 Tanner 图(传递消息的图)“展开”(即将每个迭代单独考虑)并且用乘以权重的方式将权重设置在图的“边”,则生成的解码器

收稿日期:2021-01-24

• 74 •

将是一个神经网络,可以使用基于梯度的方法进行训练。这样的“神经解码器”译码性能优于传统的置信传播译码算法,因为它们学会使用权重来减轻译码结构中短环的不利影响。神经解码器相对传统译码算法需要相对较少的迭代次数,并且在某些情况下,神经解码器误码率可以接近最大似然解码器的误码率。

然而 Nachmani 等<sup>[4]</sup>提出的方法需要大量的乘法和双曲函数运算及和内存开销,直接导致无法在实时硬件实现中有效执行这些解码器。经分析,引起神经网络解码器的高复杂度的原因如下:使用 SPA 算法作为基础置信传播译码算法,而 SPA 算法需要大量的乘法和双曲函数运算。然而发现具有乘性加权的最低和译码算法也具有出色的性能,并且与 SPA 算法相比,最低和译码算法仅采用求最小值与相加操作,在有效降低复杂度的情况下 BER 降幅最小。其次由于神经网络每层隐藏层的权重参数数量与 Tanner 图中的边缘数一致,信道码字的高密度,导致每层神经网络都存在大量参数需要学习,加大的译码的复杂度。

针对以上问题,本文提出了一种改进的低复杂度的神经网络译码算法。首先将基于循环网络的最低和算法(RNN Minsum)用于神经网络译码器,更高效,并且更容易在硬件中实现。其次分析神经网络译码器的权重参数的分布情况,有选择性地对 Tanner 图中的边缘加权,仅将被选中的边缘添加权重去训练学习,其他边缘不再加权。另外不对来自信道的对数似然比消息附加额外的参数,只在校验节点计算时添加参数,从而大大减少了模型总体的参数数量,缓解了神经网络译码器的复杂度问题。

## 1 置信传播算法与神经网络译码器

### 1.1 置信传播算法的网络表示

置信传播(BP)算法是一种高效的迭代算法,可以从 Tanner 图构造著名的 BP 解码器,Tanner 图是描述码字奇偶校验矩阵的图形表示。下面以(7,4)汉明码的检验矩阵及 Tanner 图进行举例说明,(7,4)汉明码的检验矩阵,如式(1)所示。

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Tanner 图的右半部分有  $m$  个节点,  $\{C_j | j = 1, 2, \dots, m\}$  每个节点代表一个奇偶校验约束,称为校验节点;图的左半部分有  $n$  个节点,  $\{V_i | i = 1, 2, \dots, n\}$ , 每个节点代表长度为  $n$  的码字中的一个比特位,称为变量节点;当矩阵  $\mathbf{H}$  中  $h_{ji} = 1$  时,表示第  $i$  个比特位参与第  $j$  个校验约束,对应的 Tanner 图中,变量节点  $V_i$  与校验节点  $C_j$  相连,这两个节点互称邻接节点,连接校验节点和变量节点的连线称为边缘,每个节点的邻接节点个数称为该节点的度<sup>[5]</sup>。(7,4)汉明码的 Tanner 图如图 1 所示。

在 BP 算法中,消息是通过边缘传输的。每个节点都

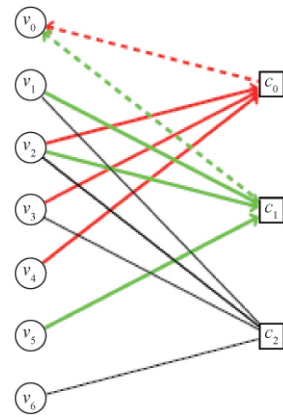


图 1 (7,4)汉明码的 Tanner 图

根据其在所相连的边缘上接收到的所有传入消息(发送边缘上接收到的消息除外)来计算其传出消息。例如,当计算  $c_0$  传递给  $v_0$  的信息,要把与  $c_0$  相连的  $v_2, v_3, v_4$  的信息纳入,但不纳入  $v_0$  传递给  $c_0$  的信息。同理计算  $v_1$  传递给  $c_2$  的信息,要把与  $v_1$  相连的  $c_1$  的信息纳入,但不纳入  $c_2$  传递给  $v_1$  的信息。

为了便于构建神经网络,Nachmani 等<sup>[4]</sup>提出一种  $L$  次迭代的 BP 译码算法的图形表示。提出的图形表示是网络结构,非常有利于构建神经网络。其中隐藏层中的处理元素对应于 Tanner 图中的边缘。用  $N$  表示代码块长度(即 Tanner 图中可变节点的数量),用  $E$  表示 Tanner 图中的边缘数目。

BP 解码器网络表示的输入层是大小为  $N$  的向量,该向量由信道输出的对数似然比(LLRs)组成。变量节点  $v$  的 LLRs 具体定义如式(2)所示。

$$L_v = \ln \frac{\Pr(C_v = 1 | y_v)}{\Pr(C_v = 0 | y_v)} = \frac{2y_v}{\sigma^2} \quad (2)$$

式中: $\sigma^2$  是信道噪声的方差; $y_v$  是观测到的接收矢量第  $v$  位。 $\{v=1, 2, \dots, n\}$ , 其中  $y_v$  是从信道中接收到的第  $v$  个码字  $C_v$  的信道输出消息。

在新的 BP 译码网络结构中,剩余的网络层除了最后一个层(即输出层)外,网络中的所有其他层(即所有隐藏层)均具有  $E$  个处理元素。对于每个隐藏层,该层中的每个处理元素产生的信息都与 Tanner 图的某个边缘传输的消息相关联。

最后,网络的最后一层(输出层)由  $N$  个处理元素组成,并输出最终的解码码字。

考虑第  $i$  个隐藏层,  $i=1, 2, \dots, 2L$ 。当  $i$  为奇数(偶数)值,该层中的每个处理元素为 Tanner 图上一个对应的从关联的变量(校验)节点到关联的校验(变量)节点的边。

第 1 个隐藏层( $i=1$ )中,与边缘  $e=(v, c)$  对应的处理元素只连接到输入层中的单个输入节点  $v$ 。在第  $i$  个( $i>1$ )隐藏层,当  $i$  为奇数值,对应于边缘  $e=(v, c)$  的处理元素连接到  $i-1$  层中所有与边缘  $e'=(v, c'), c \neq c'$  相对应的处理

元素(即所有连接到相同变量节点,但来自不同校验节点的边缘)。当  $i$  为奇数值,对应于边缘  $e=(v,c)$  的处理元素也连接到第  $v$  个输入节点。

通过网络图传输的 BP 消息计算规则如下。考虑第  $i$  个隐藏层,  $i=1,2,\dots,2L$  并令  $e=(v,c)$  是该层中处理元素的索引。用  $X_{i,e}$  来表示此处理元素的输出消息。当  $i$  为奇数(偶数)值,  $X_{i,e}$  是 BP 译码算法在  $[(i-1)/2]$  次迭代之后,从变量到校验(从校验到变量)节点生成的消息。下面给出计算方程式。

当  $i$  为奇数值,对应于边缘  $e=(v,c)$  的处理元素的输出消息如式(3)所示。

$$X_{i,e=(v,c)} = l_v + \sum_{e'=(v,c'),c' \neq c} X_{i-1,e'} \quad (3)$$

当  $i=1$  时,所有边缘  $e'$  的  $X_{0,e'} = 0$ , 因为开始时在奇偶校验节点上没有信息。

除目标边缘  $e=(v,c)$  之外,在方程式(3)中的求和采用了与变量节点  $v$  连接的的所有边缘  $e'=(v,c'),c' \neq c$ 。即在计算变量节点  $v$  传递给校验节点  $c$  的消息时,采用了与变量节点  $v$  连接的的所有边缘,除了  $v$  与校验节点  $c$  连接的那条边。这是消息传递算法的基本属性。

同理,当  $i$  为偶数值,对应于边缘  $e=(v,c)$  的处理元素的输出消息如式(4)所示。

$$X_{i,e=(v,c)} = 2 \tanh^{-1} \left( \prod_{e'=(v',c),v' \neq v} \tanh \frac{X_{i-1,e'}}{2} \right) \quad (4)$$

最后,网络输出层的第  $v$  个输出信息如式(5)所示。

$$O_v = l_v + \sum_{e=(v,c)} X_{2L,e} \quad (5)$$

2 次迭代的 BP 译码算法的图形表示如图 2 所示。

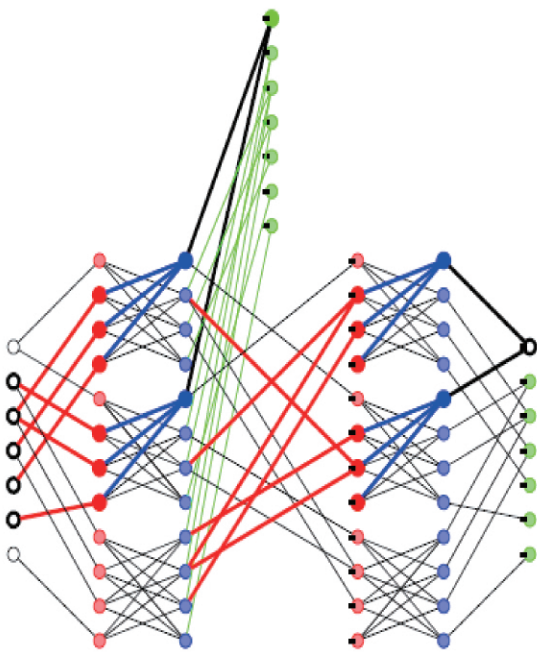


图 2 (7,4) 汉明码展开网络表示

### 1.2 基于神经网络的 BP 译码算法

对于解码器,使用与上一节相同的网络表示形式。不同之处在于,现在为 Tanner 图中的边缘分配了权重。这些权重将使用随机梯度下降训练。更准确地说,本文解码器具有与 1.1 节所述相同的网络架构,但式(3)~(5)被修改如下。

对于奇数隐藏层如式(6)所示。

$$X_{i,e=(v,c)} = \tanh(W_{i,v} l_v + \sum_{e'=(v,c'),c' \neq c} W_{i,e,e'} X_{i-1,e'}) \quad (6)$$

对于偶数隐藏层如式(7)所示。

$$X_{i,e=(v,c)} = 2 \tanh^{-1} \left( \prod_{e'=(v',c),v' \neq v} \frac{X_{i-1,e'}}{2} \right) \quad (7)$$

对于输出层如式(8)所示。

$$O_v = \sigma(W_{2L+1,v} l_v + \sum_{e=(v,c)} W_{2L+1,v,e} X_{2L,e}) \quad (8)$$

式中:  $\sigma(x) = (1 + e^{-x})^{-1}$  是激活函数,用来使最终的神经网络输出在  $[0,1]$  范围内。这使得使用交叉熵损失函数训练神经网络成为可能。可以发现,相对于式(3)~(5),如果去除了在网络的输出端添加激活函数,并将所有权重设置为 1,式(6)~(8)退化为了式(3)~(5)。因此,加入了训练参数的神经网络 BP 译码算法,其性能不会逊于普通 BP 译码算法。

容易验证所提出的基于神经网络的消息传递解码算法式(6)~(8)满足消息传递对称性条件。因此,当通过二进制无记忆对称(BMS)信道进行传输时,误码率与所传输的码字无关。因此,在训练网络参数时,可以使用单个码字再加上随机噪声就可以构建一个合格的数据集。为了方便起见,本文使用全零码字,该码字属于任何线性代码。

训练目标是训练参数  $\{W_{i,v}, W_{i,e,e'}, W_{i,v,e'}\}$  以实现让  $N$  维输出码字尽可能接近全零码字。该网络架构是一个非全连接的神经网络,本文使用随机梯度下降法来训练参数。

将使用交叉熵作为损失函数,如式(9)所示。

$$L(o, y) = -\frac{1}{N} \sum_{v=1}^N y_v \log(O_v) + (1 - y_v) \log(1 - O_v) \quad (9)$$

其中,  $O_v$  和  $y_v$  是深度神经网络的第  $v$  个比特位输出以及第  $v$  个比特位实际传输的码字(如果传输的是全零码字,则对所  $v$  的  $y_v = 0$ )。

在该网络中,输入消息和节点间传播的消息用参数加权,通过深度学习的方式获得最优值。通过神经网络优化,可以有效地减轻小环对译码的有害影响,从而提高解码性能。

但是基于神经网络的 BP 译码算法的神经网络参数非常多,并且原始 BP 算法采用的和积译码(SPA)算法需要大量的乘法和双曲函数运算,导致算法训练时间很长,甚至很难得到较好收敛效果。为了达到硬件实现的标准,必须将网络复杂度降低到可以硬件实现的水平。

## 2 降低译码复杂度的神经网络译码算法

### 2.1 基于循环神经网络的低复杂度最小和译码算法

针对上节神经译码算法存在的问题,本节通过探究上节所述神经网络译码算法的神经网络结构及参数分布情况,提出一种新的降低译码复杂度的神经网络译码算法,以降低译码算法的复杂度,并且达到硬件实现的标准,并在译码性能上不会有太大损失。以下是本文结合的方法。

#### 1) 变更基础 BP 算法

使用 SPA 作为基础置信传播算法,SPA 算法需要大量的乘法和双曲函数运算。而发现具有神经网络权重的最小和译码算法也具有出色的性能,并且与 SPA 相比,最小和译码算法仅采用求最小值与相加操作,在有效降低复杂度的情况下 BER 降幅最小。在基于最小和算法的神经网络译码算法中,式(6)及式(8)不变,而式(7)被修改为式(10),则:

$$X_{i,e=(v,c)} = \min_{e'=(v',c'),v' \neq v} |X_{i-1,e'}| \prod_{e'=(v',c'),v' \neq v} \text{sign}(X_{i-1,e'}) \quad (10)$$

最小和近似值容易产生大幅值的消息,导致误码性能下降。为了补偿这种影响,归一化的最小和(NMS)算法首先使用最小和近似来计算消息,然后使用乘以一个介于(0,1]之间的权重  $\omega$  来收缩消息,从而得出方程式,如式(11)所示。

$$X_{i,e=(v,c)} = \omega \left( \min_{e'=(v',c'),v' \neq v} |X_{i-1,e'}| \prod_{e'=(v',c'),v' \neq v} \text{sign}(X_{i-1,e'}) \right) \quad (11)$$

与上述神经 BP 解码器类似,本文为每个边缘分配可学习的权重,并训练该神经网络解码器,即归一化的最小和(NNMS)神经网络解码器。使用式(12)计算 NNMS 中的校验变量传递给变量节点的消息。

$$X_{i,e=(v,c)} = \omega_{i,e=(v,c)} \left( \min_{e'=(v',c'),v' \neq v} |X_{i-1,e'}| \prod_{e'=(v',c'),v' \neq v} \text{sign}(X_{i-1,e'}) \right) \quad (12)$$

$\omega_{i,e=(v,c)}$  是第  $i$  层中边  $(v,c)$  的权重。添加权重有两个目的:第一是校正最小和近似产生的偏差,第二是像神经 BP 解码器中的权重一样,它们消减 Tanner 图中的短环产生的影响。

NNMS 解码器和神经 BP 解码器都需要许多乘法,这通常是昂贵的运算,在硬件会要求尽量避免乘法运算。解码器可以通过偏移最小和(OMS)算法来达到不使用任何乘法的效果。与 NMS 解码一样,OMS 解码在发送消息之前先缩小消息,但之后从消息量中减去偏移量而不是乘以权重,方程式被改写,如式(13)所示。

$$X_{i,e=(v,c)} = \max_{e'=(v',c'),v' \neq v} (|X_{i-1,e'}| - \beta, 0) \prod_{e'=(v',c'),v' \neq v} \text{sign}(X_{i-1,e'}) \quad (13)$$

其中,  $\beta$  是减去的偏移量,  $\max$  函数是防止减法导致消息的正负号发生变化。本文为每个边缘分配可学习的权

重,并训练该神经网络解码器,其中校验节点计算方法如式(14)所示。

$$X_{i,e=(v,c)} = \max_{e'=(v',c'),v' \neq v} (|X_{i-1,e'}| - \beta_{i,e=(v,c)}, 0) \prod_{e'=(v',c'),v' \neq v} \text{sign}(X_{i-1,e'}) \quad (14)$$

$\beta_{i,e=(v,c)}$  是第  $i$  层中边  $(v,c)$  的权重。由于有  $\text{sign}$  等函数存在导致最小和解码器中的计算校验节点的函数并不是处处都是可微的。因此,梯度并非在所有地方都有定义。但是,这些函数仅在空间中的较低维曲线上是不可微的,并且在空间的其余部分中是可微的。

因此,本文采用标准的随机梯度下降法,使用带有扭结的激活函数,例如被广泛使用的线性整流函数(ReLU)。

#### 2) 使用循环卷积网络降低网络复杂度

由于神经网络仍有大量参数,本文使用 RNN 网络来进一步减少参数数量。使用与上述深度神经网络解码器相同的网络表示,所不同的是,在每次迭代中,将 Tanner 图中的边缘权重设置为相同的值。这种绑定将上述的前向反馈架构转换为递归神经网络架构。得到新的计算公式如下。

对于奇数隐藏层,如式(15)所示。

$$X_{i,e=(v,c)} = W_v l_v + \sum_{e'=(v',c'),c' \neq c} W_{e,e'} X_{i-1,e'} \quad (15)$$

$i=1,2,\dots,2L$  是迭代的次数。对于偶数隐藏层,如式(16)所示。

$$X_{i,e=(v,c)} = \max_{e'=(v',c'),v' \neq v} (|X_{i-1,e'}| - \beta_{i,e=(v,c)}, 0) \prod_{e'=(v',c'),v' \neq v} \text{sign}(X_{i-1,e'}) \quad (16)$$

最后,网络输出层的第  $v$  个输出信息,如式(17)所示。

$$O_v = \sigma(W_v l_v + \sum_{e'=(v',c')} W_{v,e'} X_{2L,e'}) \quad (17)$$

所提出的架构还保留了对称条件。因此使用单个代码字再加上随机噪声就可以构建一个合格的数据集。为了方便起见,使用零代码字,该代码字必须属于任何线性代码。

基于 RNN 网络构成的译码网络,相比于原始的前向反馈神经网络,将不同迭代次数的网路参数进行合并,将原始  $E \cdot (2L+1)$  的参数网络合并为规模为  $E$  的网络,有效的降低了网络复杂度。

#### 3) 选取有效的网络参数进行训练

对于 BCH(63,36) 码字,本文比较了经过训练的深度神经网络的权重,发现深度神经网络产生的权重分布在  $(-0.875, 2.201)$  范围内。验证发现,经过训练的深度神经网络中的每个隐藏层都具有接近正态分布的特性。正态分布初始化导致更好的梯度流,从而导致更好的性能。

如图3所示为 BCH(63,36) 最后一层隐藏层的权重分布直方图。

分析神经网络 BP 算法结构可知,当 Tanner 图中边缘未添加权重参数时,边缘的系数值实际上为 1,即当边缘权重为 1 时,神经网络对该边缘的译码不产生影响。图示部

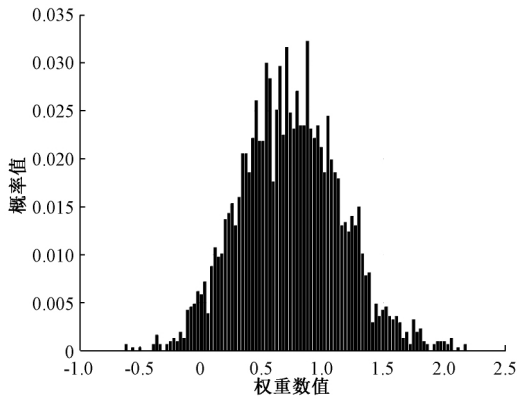


图 3 BCH(63,36)隐藏层的权重分布

分权重数值与 1 偏离很大,即网络模型对这些边缘译码过程影响较大,而大量权重值则仍保持在 1 附近,即模型对这些边缘译码过程影响较小。

另外,将式(15)和(17)中的参数  $W_v$  全部设置为 1,因为训练这些参数不会产生额外的改善。因为与从校验节点接收到的消息不同,这些来自信道的自身消息,是可靠的消息。而 Tanner 图中由于存在短环,因此从校验节点来的消息可能是不可靠的。因此,来自信道的该自消息不需要附加额外的参数,可以被照原样获取。这样可以减少大量的训练参数。

综上,本文提出一种基于 RNN 网络低复杂度最小和译码算法。首先,不对来自信道的对数似然比消息附加额外的参数,只是将参数附加于校验节点更新时的边缘上。然后,训练神经网络 BP 译码器,得到训练之后的权重分布,设立阈值  $x$ ,将权重值在  $[1-x, 1+x]$  之间的边缘判定为不需要添加权重的边缘,再次训练时将此类边缘设为 1,不加以训练。将权重值在  $[1-x, 1+x]$  之外的边缘判定为需要添加权重的边缘,然后再去训练受网络影响较大的边缘,从而得到优化的译码网络。

### 2.2 基于循环神经网络的低复杂度最小和译码算法流程

#### 1)初始化

考虑使用 BPSK 调制的通信系统。发送器发送一个随机向量  $X = \{-1, 1\}^n$ ,其中  $n$  是码字长度。从通信信道接收的随机向量是  $Y = X + Z$ ,其中  $Z$  是信道噪声向量。信道噪声是加性高斯白噪声(AWGN),计算第  $v$  个接收值的对数似然比(LLRs)

#### 2)迭代

##### (1)变量节点更新

##### (2)校验节点更新

##### (3)译码并计算损失函数

采用标准的随机梯度下降法训练,由于在神经网络最小和解码算法中使用的所有运算都是可微的,因此可以使用小批量随机梯度下降来学习偏移参数。反向传播某些构成操作  $\text{ReLU}()$ ,  $\text{min}()$ ,  $\text{abs}()$  和  $\text{sign}()$  在某些点上是不可微的。诸如 TensorFlow 之类的深度学习框架通过在不

可微的地方选择子梯度来克服这一问题,从而通过分段可微分运算进行反向传播。

#### 3)选取有效参数,进行训练

训练常规的神经网络译码器,得到训练之后的权重分布。然后,根据神经网络参数的数值选择有影响的参数,选择参数范围,设立阈值  $x$ ,将权重值在  $[1-x, 1+x]$  之间的边缘判定为不需要添加权重的边缘,再次训练网络,得到优化参数。

## 3 实验验证与结果

### 3.1 实验设计

本文以 python<sup>[6]</sup>作为实验语言,在 Tensorflow 框架下搭建实验环境。采用随机梯度下降法进行训练,最小批量大小 120,学习速率为 0.01。神经网络具有 10 个隐藏层,对应于解码算法为 5 个完整迭代<sup>[7]</sup>。

在实验时,将码字进行 BPSK 调制并加上高斯白噪声码字,通过 AWGN 信道传输零码字来创建训练数据,其信噪比范围从 1~6 dB。每个迷你批次中的在每个信噪比下都有 20 个码字(迷你批次中共有 120 个示例)。最后在在网络输出处测量解码码字中的误码率(BER)。在不同的线性代码上训练了本文的解码网络,包括 BCH(15, 11)、BCH(63, 36)、BCH(127, 106)。

在实验中,网络输出层的权重不可以加以训练,应设置为 1,否则网络将无法学习。这是由于网络可以通过更改最后一层的权重来改善损耗<sup>[8]</sup>,而无需提高其解码能力。在计算变量节点更新输出时,还会裁剪输出<sup>[9]</sup>,使其绝对值始终小于某个正的常数  $A (A < 10)$ 。这对于算法的实际(有限块长)实现也是必需的。

### 3.2 复杂度分析

假设在 Tanner 图中具有  $E$  个边缘<sup>[10]</sup>的  $(n, k)$  码字,进行  $T$  次的迭代译码。将提出的降低复杂度的循环神经网络神经网络最小和算法(LC\_RNN MS),与和积译码算法(SPA),最小和算法(MSA),基于神经网络的和积译码算法(FNSPA),基于神经网络的归一化最小和算法(FNNMSA),基于循环神经网络的和积译码算法(RNN SPA)等进行对比。译码算法复杂度对比结果如表 1 所示。

表 1 不同译码算法复杂度对比表

算法种类	乘法	双曲函数运算	网络参数个数
SPA	大量存在	大量存在	无
FNSPA	大量存在	大量存在	$(T+1)n + E(T-1)$
RNN SPA	大量存在	大量存在	$n + E$
MSA	有效减少	无	无
FNNMSA	有效减少	无	$(T+1)n + E(T-1)$
RNN MSA	有效减少	无	$n + E$
LC_RNN MS	有效减少	无	$< E$

由表1可知,对比以上各个算法发现,原始的置信传播和积译码算法具有大量乘法和双曲函数运算<sup>[11]</sup>,使用最小和译码算法,可以避免这些昂贵的操作,有效降低内存的消耗。

对比基于前向反馈神经网络<sup>[12-13]</sup>的译码算法及基于循环神经网络的低复杂度译码算法的参数分布,每一次迭代的校验节点向变量节点传递信息的参数都被设置为一样的值,导致网络参数数目从 $(T+1)n+E(T-1)$ 减少到 $n+E$ 。

综上,基于循环神经网络<sup>[14]</sup>最小和算法不仅可以减少大量的乘法操作及双曲函数运算,降低硬件实现时的资源消耗。还可以有效地减小边缘参数个数,尤其是在迭代次

数 $T$ 和边缘数 $E$ 都比较大的情况下,效果更加明显。

为了比较不同网络结构的影响,本文使用前向反馈神经网络、循环神经网络、在不选取有效参数和提取有效参数条件下分别进行训练。

首先训练常规的神经网络BP<sup>[15-16]</sup>译码器,得到训练之后的权重分布。然后选择有影响的参数,将权重值在 $[0.9, 1.1]$ 之间的边缘判定为不需要添加权重的边缘,再次训练时将此类边缘设为1,不加以训练。然后统计有效网络参数个数,对比有效参数在基于前向反馈神经网络<sup>[16]</sup>和基于循环神经网络的低复杂度算法下的个数,得出比值,具体数据如表2~4所示。

表2 BCH(15,11)码字下的参数优化

网络结构	不选取有效参数的参数个数	提取有效参数之后参数个数	减少参数比值/%
前向神经网络	$32 \times 4 + 15 \times 6 = 218$	$20 + 22 + 20 + 27 + 15 \times 6 = 179$	$179/218 = 82.1$
循环神经网络	$32 + 15 = 47$	$24 + 15 = 39$	$39/47 = 82.9$

表3 BCH(63,36)码字下的参数优化

网络结构	不选取有效参数的参数个数	提取有效参数之后参数个数	减少参数比值/%
前向神经网络	$486 \times 4 + 63 \times 6 = 2322$	$363 + 373 + 397 + 395 + 63 \times 6 = 1904$	$1904/2322 = 81.9$
循环神经网络	$486 + 63 = 549$	$383 + 63 = 446$	$446/549 = 81.3$

表4 BCH(127,106)码字下的参数优化

网络结构	不选取有效参数的参数个数	提取有效参数之后参数个数	减少参数比值/%
前向神经网络	$1008 \times 4 + 127 \times 6 = 4794$	$865 + 663 + 729 + 768 + 127 \times 6 = 3787$	$3787/4794 = 78.9$
循环神经网络	$1008 + 127 = 1135$	$763 + 127 = 890$	$890/1135 = 78.5$

如表2所示,对于BCH(15,11)码字,本文发现提取有效参数方法的参数个数比不提取有效参数方法的参数个数有效减少,参数个数减少了约17%。同理如表3所示,对于BCH(63,36)码字,参数个数减少了约18%。如表4所示,对于BCH(127,106)码字,发现参数个数减少了约22%。

对比前向神经网络结构及循环神经网络结构,循环神经网络结构参数个数大约是前向神经网络结构参数个数的22%,因此采用循环神经网络结构可以有效减少网络参数,可以很好地降低复杂度。

### 3.3 译码性能及误码率分析

观察误码曲线,如图4~6所示对于不同BCH码字,在信噪比较低时,所有译码器的误码率性能几乎相同。然而,随着信噪比越来越高,结合神经网络的译码算法由于具有可学习优化的权重,可以调整参数,克服码字短环带来的译码性能下降,体现出优秀的译码性能。同时,LC\_RNN MS的译码性能与基于

循环神经网络的最小和译码算法性能非常接近,仅仅在高信噪比的情况下,由于复杂度降低导致了译码性能的略微下降。这与网络模型降低的计算储存负担相比是可以接受的。

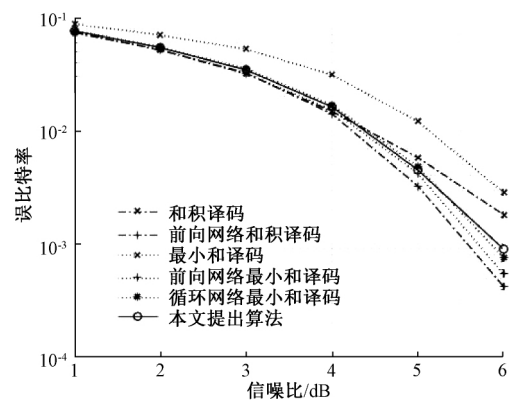


图4 BCH(127,106)误码率结果

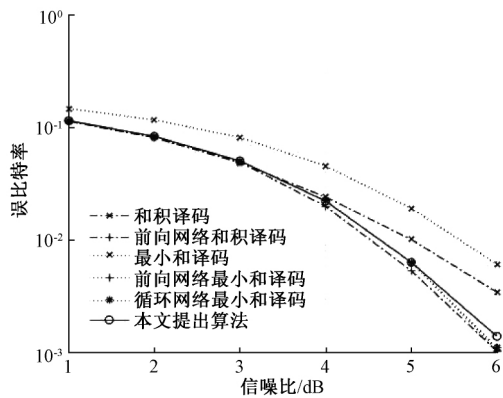


图 5 BCH(63,36)误码率结果

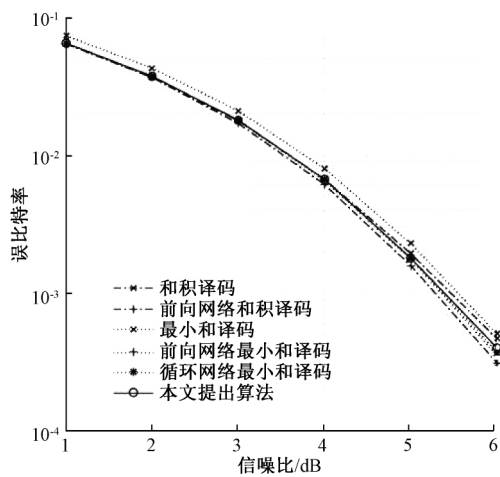


图 6 BCH(15,11)误码率结果

#### 4 结 论

本文针对神经网络译码在实际运用中存在参数过多,运算操作复杂的问题,给出了解决办法。结合 Min-sum 算法,去除了双曲函数运算并将工程实际中需要消耗大量资源的乘法操作转换为简单的加法操作。结合循环神经网络结构,将多层不同的参数约束成单层的参数。结合不同信息的属性,不对来自信道的对数似然比消息附加额外的参数,只是将参数附加于校验节点更新时的边缘上。结合边缘参数的分布,发现部分权重数值与 1 偏离很大,判定为需要添加权重的边缘。

综合以上的方法,将原有拥有众多参数,众多复杂运算的神经网络算法转变为参数数目有效减少,运算简单的新型译码算法。对比实验性能,在有效降低复杂度的情况下,译码性能损失很少,满足实际译码需求。

但与此同时,本文的译码算法对于码长较长的码字,存在训练时间长,训练难度大,译码性能差的问题,如何使网络更加容易训练,得出有效参数,提高译码网络的性能,是我们今后的努力方向。

#### 参考文献

- [1] SHANON C. A mathematical theory of communication [J]. Bell System Technical Journal, 1948,27(3):379-423.
- [2] 何鹤云. LDPC 码基础与应用[M]. 北京:人民邮电出版社,2009:123-134.
- [3] HAMMING R. Error detecting and error correcting codes[J]. Bell System Technical Journal,1950,29(3):147-160.
- [4] NACHMANI E, BE'ERY Y, BURSHTAIN D. Learning to decode linear codes using deep learning[J]. 2016 54th Annual Allerton Conf. on Communication, Control and Computing, 2016,28(7):15-17.
- [5] 魏浩源. LDPC 码的 ADMM 译码算法研究[D]. 西安:西安电子科技大学,2015.
- [6] 杨议辰,孙日明. 基于 python 的非线性方程组求解方法研究[J]. 电子测量技术, 2020,43(14):55-59.
- [7] 高林,李琪琪. 改进 BP 神经网络对期货价格的研究与应用[J]. 电子测量技术, 2019,42(23):74-78.
- [8] 贺文武,夏巧桥,邹炼. 基于变量节点更新的交替方向乘法 LDPC 惩罚译码算法[J]. 电子与信息学报, 2018,40(1):95-101.
- [9] 夏巧桥,张迪科,王泉德. 基于均衡约束的 ADMM-LDPC 译码算法[J]. 华中科技大学学报(自然科学版), 2018,46(8):77-82.
- [10] JEHOSSUA B. Neural networks, error-correcting codes, and polynomials over the binary n-cube[J]. IEEE Transactions on Information Theory, 1989, 35(5):976-987.
- [11] 姚品,万旺根. 基于深度学习和属性特征的行人再识别算法[J]. 电子测量技术,2020,43(12):70-74.
- [12] 柏春燕,谢显中,王新梅. 基于前馈神经网络的分组码译码方案[J]. 西安电子科技大学学报,1999,26(2):160-164.
- [13] 王立刚,张志佳,李晋,等. 基于卷积神经网络的 LED 灯类字体数字识别[J]. 电子测量与仪器学报,2020,34(11):148-154.
- [14] 王永利,曹江涛,姬晓飞. 基于卷积神经网络的 PCB 缺陷检测与识别算法[J]. 电子测量与仪器学报,2019,33(8):78-84.
- [15] 李禹剑,李剑,辛伟瑶. 一种基于 BP 神经网络的老旧桥梁健康诊断方法[J]. 国外电子测量技术,2020,39(2):19-22.
- [16] 吴睿曦,肖秦琨. 基于深度网络和数据增强的多物体图像识别[J]. 国外电子测量技术, 2019,38(5):86-90.

#### 作者简介

孙浩杰,硕士研究生,主要研究方向为 LDPC 译码。

E-mail: 206716123@qq.com