Polar 码软件实现方案研究

闫宁[1] 王涛[2] 刘鑫[3] 胡睿[4]

1 安徽新华学院，安徽合肥，230088；

2 奇瑞汽车(合肥)研发院，安徽合肥，230051；

3 安徽新华学院，安徽合肥，230088；

4 阜阳师范大学，安徽阜阳，236041；

摘要：信道编码在通信系统中至关重要，它能确保信息传输质量。极化码作为一种新型信道编码技术，性能接近香农极限，在无线通信、数据存储等领域具有巨大应用潜力。传统线性分组码在码长增加时，译码复杂度呈指数级增长，限制了其实用性。相比之下，极化码因编译码复杂度较低而更具优势，满足 5G 中控制信道编码的要求，未来有望在通信系统中继续发挥重要作用。然而，极化码的生成矩阵实现过程需多次申请空间，花费较多时间，且译码实现方法判断过程繁琐，存储空间不够优化。为克服这些缺陷，本文提出了一种极化码优化的编译码方法。该方法旨在改进极化码的编译码过程，提高实现效率和优化存储空间，以进一步提升极化码在实际应用中的性能和实用性，为通信系统的未来发展提供有力支持。

关键词：极化码；软件实现；编译码

DOI:10.69979/3041-0673.25.03.047

引言

在当今这个信息爆炸的时代，无线通信技术正以前所未有的迅猛之势蓬勃发展，从早期的 2G 到如今广泛普及的 5G，每一次技术的迭代都为人们的生活带来了翻天覆地的变化[1]。而极化码作为一种新兴且具有优异性能的信道编码技术，正逐步展现出其在通信领域的广泛应用潜力[2]。作为一种基于信道极化现象的编码方法，极化码在理论上能够达到香农容量限，为无线通信系统的高效率、高可靠性传输提供了新的可能。因此，深入研究极化码的软件实现方案，不仅对于推动无线通信技术的发展具有重要的理论意义，同时也具备极高的实际应用价值。通过不断优化极化码的编码与解码算法，可以进一步提升软件的运行效率和可扩展性，从而使其更好地适应各种复杂的应用场景和需求。未来，极化码有望在更多领域得到广泛应用，为无线通信技术的发展注入新的活力。

1 极化码编译码

1.1 极化码概述

极化码是根据信道极化现象提出的一种信道编码方式。极化码构造的核心是通过信道极化（channel polarization）处理，在编码侧采用方法使各个子信道呈现出不同的可靠性。当码长持续增加时，部分信道将趋向于容量近于 1 的完美信道（无误码），另一部分信道趋向于容量接近于 0 的纯噪声信道。选择在容量接近于 1 的信道上直接传输信息以逼近信道容量，是目前唯一能够被严格证明可以达到香农极限的方法。极化码的编译码器设计采用了递归的方法，这种方法不仅简化了编码与解码的流程，还使得极化码的实现复杂度相对较低，同时占用的系统资源也较少。通过递归结构，极化码能够有效地利用信道极化的特性，将信息比特分配到更可靠的子信道上，从而提高通信的可靠性。这种设计不仅优化了编码效率，还确保了极化码在实际应用中的可行性和经济性。

1.2 极化码编译码原理

极化码的编码与译码机制核心在于信道极化理论的应用。该理论表明，通过特定的编码策略，使信道产生两级分化的特性[3]。详细来说，采用巧妙的编码方式使一类信道趋于完全可靠的信道，其容量逐渐增大以至于趋近于 1，使其成为近似无噪声干扰的信道；另一类信道则会趋向于完全不可靠的(纯噪声)信道，其信道容量不断减小，直至趋近于 0，使其成为近乎纯噪声的信道。正是基于这一理论，极化码能够在资源受限的条件下，达成高传输速率与低误码率的目标。

1.3 极化码编译码软件实现现状

编码方面：极化码生成矩阵的构造方法有多种，但其中一种常见且有效的方法是利用克罗内克乘积。这种方法通过多次应用克罗内克乘积，从一个较小的基本矩阵（如 F 矩阵）出发，逐步构造出所需的极化码生成矩阵。由于克罗内克乘积每次都会生成一个新的矩阵，且这个矩阵的维度是原始矩阵维度的乘积，因此随着乘积次数的增加，所需的空间也会急剧增加[4]。同时，由于每次乘积都需要进行大量的计算，因此时间花费也会相应增加。

译码方面：在译码实现方法所采用的节点消息计算公式选择中，每个节点仅依据其序号的奇偶性来进行判断[5]。这种方法要求每个节点都必须存储所有其他节点的信息，从而导致在实际实现过程中，判断步骤相对复杂且所需的存储空间未能得到有效优化。

2 极化码编码软件实现方案

针对现有技术中存在的缺陷与不足，本文提出了一种优化的编码方案。该方案在计算生成矩阵时，摒弃了传统的克罗内克乘积方法，转而采用循环复制的策略来构建生成矩阵。这一改进不仅有效避免了空间的重复分配，还显著节省了计算所需时间，从而提升了整体效率。编码的生成矩阵公式为 $G_N = B_N F^{\otimes n}$，公式中 $F^{\otimes n}$ 表示完成比特反序前的生成矩阵，$F^{\otimes n} = F \otimes F^{\otimes(n-1)}$，$F = \begin{vmatrix} 1 & 0 \\ 1 & 1 \end{vmatrix}$，其中 N 为码长，$n = \log_2 N$，$\otimes$ 为矩阵的克罗内克乘积，B_N 是排序矩阵用以完成比特反序重排操作；当码长 N 较长时，为了不需要重复 $n = \log_2 N$ 次克罗内克乘积计算和对应的软件存储空间开辟，本文方案仅需要单次申请存储空间大小为 $N \times N$ 的矩阵，然后进行循环复制生成生成矩阵，软件计算过程如下：

1）将申请的 $N \times N$ 维矩阵左上角复制成 $F = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$；

2）将 $N \times N$ 维矩阵（2^{index}，0）坐标位置复制成 $F = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$、（0，$2^{index}$）位置复制为 $\mathbf{0} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$、（$2^{index}$，$2^{index}$）位置复制成 $F = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$，会在 $N \times N$ 矩阵左上角形成 $F^{\otimes 2}$，$index$ 表示一个计数变量，范围是 $1 \le index \le log_2 N$，首次操作时令 $index = 1$，之后每循环一次加 1；

3）以 $F^{\otimes 2}$ 为复制单位操作 2，得到 $F^{\otimes 3}$；

4）重复步骤 2）、3），直到循环到 $index = \log_2 N$，得出 $F^{\otimes n}$；

5）对 $F^{\otimes n}$ 进行比特反序重排得到生成矩阵，对信息比特进行编码得到码字，完成极化码编码器工作；

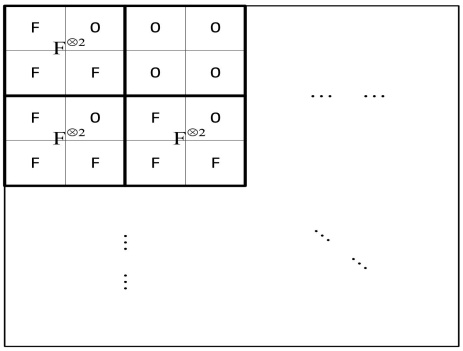

图 1 生成矩阵中的 $F^{\otimes n}$ 示意图

图 1 是生成矩阵中的 $F^{\otimes n}$ 示意图。其中：$F^{\otimes 2}$ 是计算第一次得到的结果，然后以它为单位再进行循环操作得到 $N \times N$ 维矩阵 $F^{\otimes n}$。

3 极化码译码软件实现方案

译码算法采用串行抵消译码，算法中 LLR（对数似然比）的计算是采用通过递归完成， 算法中定义函数 f 和 g 如下：

$$f(a,b) = \ln(\frac{1+e^{a+b}}{e^{a}+e^{b}}) \quad (1)$$

$$g(a,b,u) = (-1)^u a + b \quad (2)$$

上公算法中 $a,b \in R$，$u \in \{0,1\}$，a,b,u 分别表示为 f 和 g 的自变量，LLR 的递归运算用公式表示如下：

$$L_N^{2i-1}(y_1^N, \hat{u}_1^{2i-2}) = f(L_{N/2}^i(y_1^{N/2}, \hat{u}_{1,o}^{2i-2} \oplus \hat{u}_{1,e}^{2i-2}), L_{N/2}^i(y_{N/2+1}^N, \hat{u}_{1,e}^{2i-2})) \quad (3)$$

$$L_N^{2i}(y_1^N, \hat{u}_1^{2i-1}) = g(L_{N/2}^i(y_1^{N/2}, \hat{u}_{1,o}^{2i-2} \oplus \hat{u}_{1,e}^{2i-2}), L_{N/2}^i(y_{N/2+1}^N, \hat{u}_{1,e}^{2i-2}), \hat{u}_{2i-1}) \quad (4)$$

上述公式（3）、（4）中 u^i 为码字中第 i 的码元，$\hat{u}^i$ 为第 i 码元的译码估计值，$\hat{u}_1^N$ 为 $\hat{u}_1,\ldots,\hat{u}_N$，$y_1^N$ 为译码器接收到的消息 $y_1,\ldots,y_N$，$\hat{u}_{1,o}^{2i-2}$ 表示 1 到 $2i-2$ 奇数索引的子向量，公式中下标 o 表示奇数，$\oplus$ 表示亦或，$\hat{u}_{1,e}^{2i-2}$ 表示 1 到 $2i-2$ 偶数索引的子向量，公式中下标 e 表示偶数，$L_N^{2i-1}(y_1^N, \hat{u}_1^{2i-2})$ 为根据接收信号 y_1^N 与估计序列 $\hat{u}_1^{2i-2}$ 计算码元 u_{2i-1} 的对数似然比消息，$L_N^{2i}(y_1^N, \hat{u}_1^{2i-1})$ 为根据接收信号 y_1^N 和估计序列 $\hat{u}_1^{2i-1}$ 计算码元 u_{2i} 的对数似然比消息；

根据公式（3）、（4）的递归运算公式采用树形结构计算每个节点的对数似然比消息，进行软件译码计算，过程如下：

1）首先建立层数为 $\log_2 N$ 的满二叉树，依照译码的码元序号 i，从树的第一层开始计算；

2） 符号 $i\%2$ 表示 i 对 2 取余，当 $i\%2 = 1$ 时本层

节点消息选择公式（4）计算；当$i\%2=0$时则选择公式（3）计算；

3）令$i=i/2$，计算下一层即第二层；

4）重复上述计算过程中的步骤 2）、3）直到计算完树的第$\log_2 N$层节点消息结束计算过程。

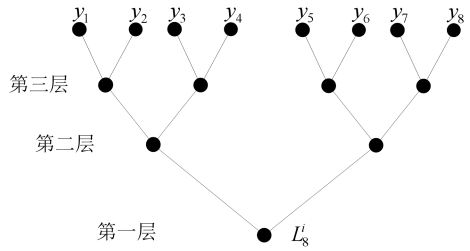

图 2 树结构实现译码的示意图

图 2 是以树结构实现译码的示意图。其中：选择码长N=8 为例，对第i码元译码，树含有$\log_2 8=3$层，$y_1,\ldots,y_8$为译码器接收到的信息。

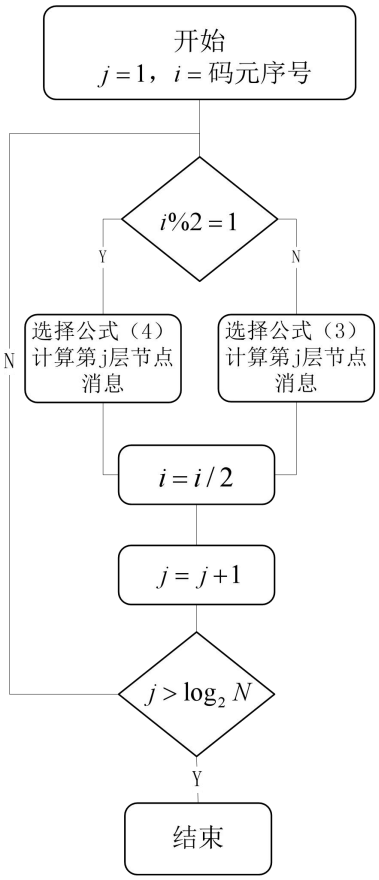

图 3 计算树节点消息的流程示意框图

图 3 是计算树节点消息的流程示意框图。其中：j为树的层数，一直计算完最后一层$\log_2 N$结束循环。

4 结束语

在本文中，极化码编码方面我们实现了一种创新的生成矩阵构建方法，该方法显著地避免了传统方法中多次使用克罗内克乘积的繁琐步骤。更重要的是，我们无需在构建过程中反复开辟新的存储空间，从而有效优化了内存使用并提升了整体效率。这一改进不仅简化了生成矩阵的计算流程，还显著降低了资源消耗，同时避免多次使用克罗内克乘积的计算为实际应用中的编译码过程带来了显著的性能提升，存储空间由$2(N^2-1)$减少为N^2；在极化码译码实现方面采用树结构方式计算节点的对数似然比消息，对下一码元译码时只需重新计算树节点的信息，存储空间由$N\log_2 N$减少到$N-1$；在采用树结构方式基础上提出的公式选择方法，使每一层节点计算公式相同，而不是每计算一个节点需要根据公式（3）、（4）上表去计算使用哪个译码计算公式，较大程度上简化了译码消息的计算过程，而其他改进的译码算法都建立在逐次消除算法上；随着码长的增加，编码译码有益效果越显著。

综上所述，本研究紧密围绕未来发展趋势，明确了深化技术研究、优化软件性能及拓宽应用场景的核心方向。尽管本研究取得阶段性成果，但仍需持续关注前沿技术，不断完善软件解决方案。期待后续研究能在现有基础上，进一步突破技术瓶颈，拓展应用边界，为行业发展注入新活力，为社会创造更大价值。

参考文献

[1]袁文韬.研究5G移动通信技术在物联网的应用[J].中国新通信，2020,1(1).

[2]赵坤.极化编码调制技术研究[D].北京:北京邮电大学,2020.

[3]崔茵,袁辽,倪卫明.高斯信道下极化码的子信道错误概率计算[J].微型电脑应用，2017，33(2):5.

[4]胡利港.极化码缩短核矩阵构造方法研究[D].广州:暨南大学,2022.

[5]王燕,刘顺兰,奚珍珍,等.一种基于增强奇偶校验码级联极化码的新型编译码方法[J].电信科学,2021,037(003):125-132.

作者简介：闫宁（1992），女，汉，山东邹城，安徽新华学院，助教（山东大学硕士），研究方向：信号处理/通信/软件。

项目名称：2024 省级大创 （项目编号：S202412216216）；

项目名称：安徽省高等学校科学研究项目（自然科学类）-基于多源数据的长三角城市群宜居度的时空特征研究（项目编号：2023AH051811）