 DOI: https://doi.org/10.12349/tie.v3i9.11782

Performance Analysis of Tail-Biting and Zero-Tail Turbo Codes with Parallel Decoding in Scattering Channels

Jingyuan Zhao Hao Meng

The 54th Research Institute of CETC, Shijiazhuang, Hebei, 050081, China

Abstract

Severe multipath fading in scattering channels demands robust anti-fading capabilities from communication equipment, making error-correcting coding essential for maintaining transmission reliability. As Turbo codes are widely deployed, their termination strategy significantly dictates both coding efficiency and decoding complexity. This study evaluates the parallel decoding performance of Tail-Biting (TB) versus Non-Tail-Biting (NTB) Turbo codes across various code lengths and parallel degrees in scattering environments. We implement parallel processing using the Max-Log-Map algorithm with uniform block partitioning and boundary state exchange. Results show that for short codes (e.g., N=128), TB codes outperform NTB counterparts by approximately 1 dB, primarily due to the elimination of tail-bit overhead and uniform gain distribution, which also mitigates the high-SNR error floors typical of NTB codes. As code length increases, the boundary-free advantage of TB codes persists, yielding a 0.5 dB gain at N=4096. However, this gap vanishes at N=8192 as structural differences become negligible relative to the sequence length. Crucially, we observe distinct sensitivities to parallelism: TB performance degrades in short-code, high-parallelism scenarios due to insufficient sub-block convergence, whereas NTB codes remain robust, with performance largely dictated by channel conditions rather than parallel granularity. These findings offer practical guidelines for jointly optimizing termination methods and decoding architectures in scattering communication systems.

Keywords

troposcatter communication; Turbo; zero-tail; tail-biting; boundary state exchange

散射信道中咬尾与非咬尾 Turbo 码并行译码性能研究

赵靖远 孟颢

中国电子科技集团公司第五十四研究所，中国 · 河北 石家庄 050081

摘　要

散射通信由于存在较为严重的多径衰落，对通信设备的抗衰落能力提出更高要求，信道纠错编码作为无线通信中的重要部分，是提高散射信道通信传输性能的重要手段，Turbo码作为目前广泛使用的编码方式，它的终止方式将影响编码效率和译码实现的复杂度。本文针对散射信道环境，系统对比了咬尾和非咬尾Turbo码在不同码长、不同并行度下的并行译码性能，采用Max-Log-Map算法通过均匀分块与边界状态交换实现并行处理。仿真结果表明：在短码长（如N=128）下，咬尾码凭借无拖尾比特损失和均匀的译码增益，相比非咬尾码拥有约1dB的性能优势，且有效抑制了非咬尾码在高信噪比下的误码平层；随着码长增加，咬尾码的无边界优势逐渐凸显，在N=4096时相比非咬尾码获得约0.5dB的净增益，而在长码（N=8192）下两者性能趋于一致。此外，并行度对两种结构的影响显著不同：咬尾码性能对子段长度敏感，短码高并行度易导致收敛失效和平层出现，而非咬尾码对并行度变化表现出较强的鲁棒性，其性能主要受限于信道条件而非并行粒度。该研究为散射信道系统中Turbo码的编码方式与译码架构的联合优化提供了参考依据。

关键词

散射通信；Turbo码；咬尾；非咬尾；边界状态交换

1 引言

散射通信利用对流层对电磁波的散射效应实现超视距传输，具有抗干扰能力强、部署灵活等优势，在军事通信、应急保障及偏远地区覆盖等场景中具有不可替代的作用。但是在散射信道存在严重的多径衰落，对通信能力提出了更高的要求[1]。信道纠错编码在通信过程中能够有效提升传输性能，其中 Turbo 码凭借其接近香农限的译码性能和灵活的迭代结构，已成为散射通信系统中的主流方案之一。在实际工程中，Turbo 码的终止方式对编码效率、译码性能以及并行实现的复杂度有直接影响。非咬尾（zero-tailed）结构通过在编码器末尾添加尾比特强制回归全零状态，实现简单但引入了额外的码率损失；咬尾（tail-biting）结构则通过约束编码器起始状态与结束状态相等，消除了尾比特开销。

散射通信业务呈现多样化特征，战术短报文、控制信

【作者简介】赵靖远（1994–），女，中国河北石家庄人，硕士，工程师，从事信道纠错编码研究。

令等需要短帧快速传输，图像、视频等业务则依赖于长帧高吞吐 [2]。在散射信道这一特定场景下，码长跨度对两种终止方式性能敏感度的系统性影响尚缺乏深入评估。此外，随着并行译码技术在低时延应用中的广泛应用，咬尾结构的环形边界交换机制与非咬尾结构的链式单向交换在收敛特性、实现复杂度及不同码长下的适应性差异仍需定量分析。本文针对散射信道环境，在 128~8192 的宽码长范围内系统对比分析咬尾与非咬尾 Turbo 码的译码性能。采用 MAX-LOG-MAP 算法实现分量码译码，通过均匀分块与边界状态交换的实现并行化处理，对两种终止方式在不同码长、不同并行度下的误码率性能、收敛特性及复杂度进行评估。

2 Turbo 码的基本原理

本文后续仿真中采用的是 LTE-Turbo 码 [3]，采用 3 个寄存器的系统递归卷积码作为分量码，编码器中的交织器采用的是二次置换多项式交织器（Quadratic Permutation Polynomial，QPP），编码结构如图 1 所示。

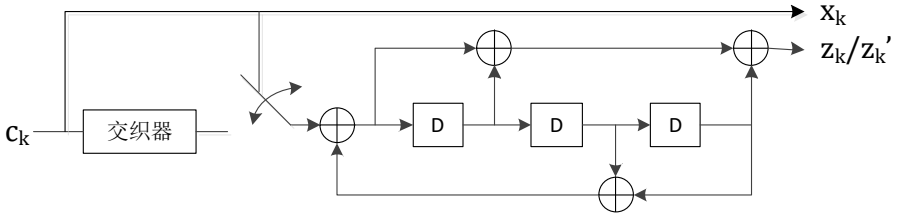


图 1 LTE–Turbo 码编码器结构

设编码器的待编码信息长度为 N，输入编码器的信息比特序列为 $c_0,c_1,...,c_{N-1}$，编码器的输出包括三部分，信息位 $x_0,x_1,...,x_{N-1}$，校验位包括顺序编码的校验位 $z_0,z_1,...,z_{N-1}$，以及信息交织后编码的校验位 $z'_0,z'_1,...,z'_{N-1}$，所以该编码器得到的母码码率为 1/3。

交织器作为 Turbo 码的重要组成部分，直接关系到 Turbo 码的译码性能的优越性，该编码器中采用的 QPP 交织器，具有代数结构清晰，复杂度低，硬件友好并且支持并行译码等优点被应用于 LTE 标准中，其映射函数如公式 1 所示：

$$\Pi(k)=(f_1k+f_2k^2)mod\ N \tag{1}$$

其中，k 表示交织前的数据位置，k=0,1,2...,N−1，Π(k) 为交织后的数据位置，f_1 和 f_2 代表交织参数，由交织长度 N 决定。

2.1 非咬尾编码

对于图 1 的编码器，若采用非咬尾的终止方式编码，编码过程为：（1）待编码信息顺序输入计算校验位 z_k；（2）待编码信息交织后输入计算校验位 z_k ‘；（3）添加尾比特，将编码器状态回 0。

非咬尾编码码字为系统码字，信息位在前，校验位在后，串行输出格式如下：

$$x_0x_1x_2...x_{N-1}z_0z_0'z_1z_1'z_2z_2'...z_{N-1}z_{N-1}'$$

尾比特需要传输的有归零的尾比特 x_N、x_{N+1}、x_{N+2} 和其产生的校验位 z_N、z_{N+1}、z_{N+2}，以及交织部分的归零尾比特 x_N'、x_{N+1}'、x_{N+2}' 和其产生的校验位 z_N'、z_{N+1}'、z_{N+2}'。所以输出的编码码字串行最终输出为：

$$x_0x_1x_2...x_{N-1}x_Nx_{N+1}x_{N+2}x_N'x_{N+1}'x_{N+2}'z_0z_0'z_1z_1'z_2z_2'...$$
$$z_{N-1}z_{N-1}'z_Nz_N'z_{N+1}z_{N+1}'z_{N+2}z_{N+2}'$$

则该编码器非咬尾编码后的最终长度为 3N+12bit，实际码率为 $R_{actual}=\frac{N}{3N+12bit}$，由此可见，非咬尾编码会带来码率损失，码长越短，尾比特占比越大，效率越低。

在已知最终状态情况下可以直接通过查表得到尾比特及产生的校验位，具体查表内容如下表：

表 1 尾比特产生表

当前状态	归零信息位	归零校验位
000	000	000
001	100	000
010	110	100
011	010	100
100	011	110
101	111	110
110	101	010
111	001	010

2.2 咬尾编码

由图 1 编码器结构可以得出编码器状态变化和输入比特之间的关系。k 时刻的输入信息为 c_k，状态为 $d_{k,0}$、$d_{k,1}$、$d_{k,2}$，

$$\begin{aligned} d_{k+1,0}&=c_k+d_{k,1}+d_{k,2}\\ d_{k+1,1}&=d_{k,0}\\ d_{k+1,2}&=d_{k,1}\end{aligned} \tag{2}$$

可得到状态转移矩阵

$$G=\begin{bmatrix}0&1&1\\1&0&0\\0&1&0\end{bmatrix}$$

k 时刻状态转移至 k+1 时刻状态可以表示为 $S_{k+1}=GS_k+C_k$，其中 $C_k=(c_k\quad 0\quad 0)^T$。

当初始状态为 S_0，终止状态为 S_N，那么反复代入可以得到

$$S_N=G^NS_N+\sum_{k=1}^{N}G^{N-k}C_k \tag{3}$$

进行咬尾状态计算，则令 $S_N=S_0=S_c$，可推导得到

$$S_c=[I+G^N]^{-1}\sum_{k=1}^{N}G^{N-k}C_k \tag{4}$$

$\sum_{k=1}^{N}G^{N-k}C_k$则是当$S_0=(0,0,0)$时得到的终止状态，当 N 不为 7 的倍数，也就是$[I+G^N]^{-1}$存在时，即可得到咬尾状态 S_c。

咬尾码字的编码过程为：（1）预编码，将编码器的初始状态设置为全零状态 (0,0,0)，进行编码，得到终止状态 S_c，再根据终止状态和码块长度 N 查找表得到编码器的循环状态；（2）正式编码，将编码器的初始状态设置为预编码阶段得到的循环状态再进行一次编码，编码器的输出就是最终的编码。

输出码字的长度为 3N，串行格式为：

$$x_0x_1x_2...x_{N-1}z_0z_0'z_1z_1'z_2z_2'...z_{N-1}z_{N-1}'$$

2.3 Max–Log–Map 译码算法

Max-Log-Map 算法是 Turbo 码及卷积码迭代译码的核心算法之一，本质上是对最大后验概率（MAP）准则在对数域的简化近似。其核心思想在于通过前向递推和后向递推两个过程，递归的计算每一时刻的对数似然比（LLR），以此来判断每个接收信息最可能是“0”还是“1”。为了在工程实现上尽可能的降低复杂度，Max-Log-Map 利用对数运算将概率的乘积关系转化为加法关系，将对数域中的指数求和的复杂运算近似为简单的求最大值，这一处理虽然会相比较 MAP 算法带来性能损失，但根据仿真结果和工程实践经验，可以在译码运算中增加校正项有效提升性能，使其在保证性能的同时，并降低对硬件实现的要求。

在第 k 时刻，接收信息的后验概率对数似然比 $L(c_k)$ 定义为：

$$L(c_k)=ln\frac{P(c_k=1|Y_1^N)}{P(c_k=0|Y_1^N)} \tag{5}$$

Y_1^N为 k 时刻接收到的信息序列，根据贝叶斯规则，式（5）可以改写成

$$\begin{aligned}L(c_k)&=ln\frac{\dfrac{P(c_k=1,Y_1^N)}{P(Y_1^N)}}{\dfrac{P(c_k=0,Y_1^N)}{P(Y_1^N)}}\\&=\ln\frac{\sum_{c_k=1}^{(s',s)}P(S_{k-1}=s',S_k=s,Y_1^N)}{\sum_{c_k=0}^{(s',s)}P(S_{k-1}=s',S_k=s,Y_1^N)}\end{aligned} \tag{6}$$

(s',s)表示在 k 时刻由信息符号 $c_k=i$ 引起的状态转移集合。如果将接收序列Y_1^N按照当前 k 时刻分成三部分，当前接收到的为 y_k，当前时刻之前接收到的为序列$Y_1^{k-1}=(y_1,...,y_{k-1})$，当前时刻之后接收到的为序列$Y_{k+1}^N=(y_{k+1},...,y_N)$，可得

$$\begin{aligned}P(S_{k-1}=s',S_k=s,Y_1^N)&=P(s',s,Y_1^{k-1},y_k,Y_{k+1}^N)\\&=P(s',Y_1^{k-1})P(s,y_k,Y_{k+1}^N|s',Y_1^{k-1})\\&=P(s',Y_1^{k-1})P(s,y_k|s')P(Y_{k+1}^N|,s)\\&=\alpha_{k-1}(s')\gamma_k(s',s)\beta_k(s)\end{aligned} \tag{7}$$

式（7）中的 α 定义为前向度量，β 为后向度量，γ 为分支度量。咬尾码和非咬尾码在并行译码架构中的主要区别就是在每次迭代对前向度量和后向度量初始化有不同，对边界值的交换方式有所不同，下面对两种不同终止方式的并行译码结构进行说明。

并行译码是为了降低译码时延，将接收到的序列均匀划分为 W 个子块并行处理，各子块独立进行 Max-Log-Map 译码，在块边界对度量信息进行交换。

2.3.1 非咬尾并行译码

非咬尾码字在编码结束后，由于尾比特的归零运算，可以认为编码器的初始状态和最终状态都为 0，这对于译码器是已知的，所以对于首子块的前向度量初始化为

$$\alpha(s)_0=\begin{cases}0，&s=零状态\\\infty，&s\neq零状态\end{cases} \tag{8}$$

尾子块的后向度量初始化为

$$\beta(s')_{W+1}=\begin{cases}0，&s=零状态\\\infty，&s\neq零状态\end{cases} \tag{9}$$

中间子块通过迭代交换边界状态。由于首尾状态被编码约束锁定，边界交换呈链式单向结构，如图 2 所示。

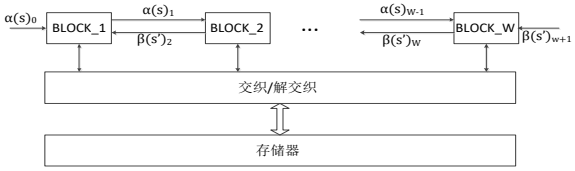


图 2 非咬尾码并行译码边界值交换示意图

对于 BLOCK_1 和 BLOCK_W，由于非咬尾编码的初始状态和最终状态都是 0，所以对于任意每次迭代的初始值都是固定的。而对于中间块，第一次迭代时的边界状态时未知的，可以认为是等概的，一般可以都设置为 0，到了后续迭代时，利用上一次迭代相邻块的前后向度量的终值作为本次迭代的边界初始值。

2.3.2 咬尾并行译码

咬尾码字的初始状态和终止状态虽然相同，但是对于译码器来说该状态值是未知的，所以在第一次迭代时，将所有子块的前向 / 后向度量均初始化为等概率分布，迭代过程中，各子块结束状态传递至下一子块作为起始状态，尾子块结束状态回传至首子块，形成环形闭合，如图 3 所示。

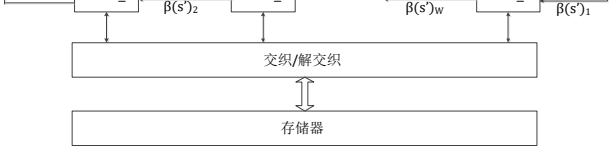


图 3 咬尾码并行译码边界值交换示意图

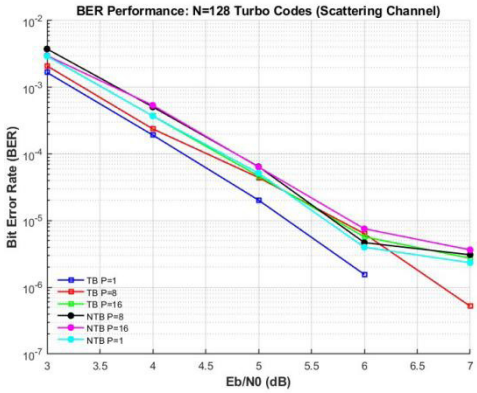
3 仿真结果与分析

为了验证咬尾与非咬尾终止方式下并行译码算法在散

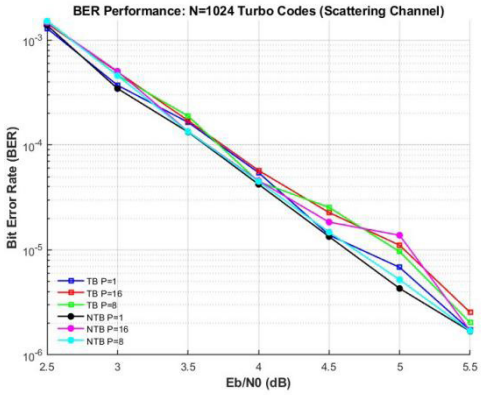
射信道中的性能差异，本节基于 MATLAB 平台进行了系统仿真。信道模型采用准静态瑞利衰落模型，假设各径独立同分布。仿真重点考察了不同信息位长度 128、1024、4096、8192 及并行度 1、8、16 下的误码率（BER）性能。仿真条件设置如表 3 所示，仿真结果如图 4 所示。

表 3 仿真条件设置

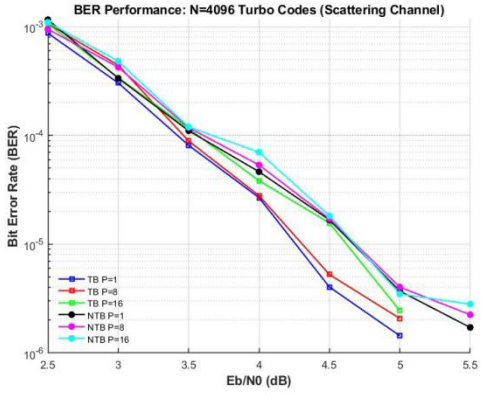
调制方式	BPSK
散射信道设置	多径数 =6，最大多径时延 =5 个符号周期
信息比特长度	128,1024,4096,8192
码率	1/3
译码算法	Max-Log-Map



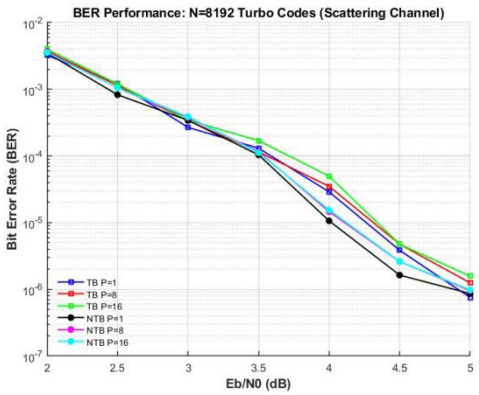
(a) N=128



(b) N=1024



(c) N=4096



(d) N=8192

图 4 咬尾和非咬尾码字 BER 仿真图

由图 4 仿真结果可以得到以下结论：

在短码 N=128 情况下，由于非咬尾码字归零比特占比较高造成码率损失，咬尾码比非尾码存在 1dB 左右的优势，并且非咬尾码明显在高信噪比下出现误码平层，在并行度为 16 时，由于每段长度仅有 8bit 收敛速度降低，咬尾码字同样出现了误码平层；

在中短码长 N=1024 下，两者性能相当，非咬尾码表现出微弱的 0.3dB 优势，这可能是由于下边界效应的统计波动及非咬尾码强制终止状态在特定信道下的偶然增益所致，随着码长增加至 N=4096，咬尾码的无边界结构优势逐渐显现，相比之下，非咬尾码在长码下因强制终止状态导致的边缘效应累积，使其性能逐渐落后，所以在 N=4096 时相比 非咬尾码获得了约 0.5dB 的净增益；

在长码 N=8192 时，由于码长较长，非咬尾码强制归零带来的码率损失和边界效应被显著稀释，同时，咬尾码的无边界优势在长序列内部比特中也未能体现出显著差异。因此，两者整体 BER 性能相近，并行度影响也变得微弱，每个子码段的长度足够长，保证了迭代译码的充分收敛和良好的伪随机性。

整体来看，并行度对非咬尾码字的影响较低，可能是由于非咬尾码的强制归零提供了稳定的局部译码约束，使得译码性能主要取决于码长和信道条件，而非并行处理的粒度。相比之下，咬尾码的性能对并行度更为敏感。在短码高并行度下，咬尾码因段长过短出现误码平层，在长码下，咬尾码码性能趋于稳定。

4 结语

本文针对散射信道环境，系统评估了咬尾与非咬尾 Turbo 码在并行译码架构下的性能表现。通过涵盖多码长与多并行度的广泛仿真分析，本研究明确了两种终止机制在复杂衰落环境下的性能敏感度差异。研究结果表明，终止方式与并行度的组合策略需依据具体码长进行精细化权衡。在短码约束下，咬尾码通过消除拖尾比特损失提供了约 1dB 的性能增益，是高可靠性短包传输的首选；而在长码并行处理

中，非咬尾码因其强制终止机制提供的稳定局部约束，对并行度变化不敏感，更适合构建高吞吐量的并行译码硬件流水线。相比之下，咬尾码在长码下性能优异，但其对子段长度的敏感性要求在实际设计中要保证足够的并行分支长度以避免收敛失效。这为散射通信系统中 Turbo 码终止方式的选择及编码参数优化提供了重要的理论依据与设计准则。后续工作可以在以下两个方向继续探索：（1）高码率场景下的性能极限研究：深入探究在更高码率条件下，咬尾与非咬尾结构在散射信道中的性能权衡及错误平台效应；（2）面向硬件实现的低时延优化：结合 FPGA 平台，针对并行译码架构进行资源与流水线优化，以实现更低时延、更高吞吐量的实际部署。

参考文献

[1] 张涛，刘莹，孙柏昶，等. 对流层散射通信及其应用［M］. 北京：电子工业出版社，2020.

[2] 李志勇，秦建存，梁进波. 对流层散射通信工程[M]. 北京：电子工业出版社，2017.

[3] 张功岑.基于FPGA的LTE-A系统的Turbo编译码算法的并行化研究与实现[D].电子科技大学,2014.

[4] 平磊，面向5G通信的咬尾卷积码和Turbo码技术研究[D],西安电子科技大学，2017.

[5] H. T. Pai, “Sliding Window Algorithm for TBCC Decoding,” in IEEE Communications Letters,vol. 14, no. 12, pp. 1158-1160, December 2010.

[6] 3GPP. Evolved Universal Terrestrial Radio Access（E-UTRA）; Multiplexing and channel coding：TS 36.212 V15.2.0[S].2018.

[7] 汪祖海，短块长极低码率信道编码优化设计[D],北京邮电大学，2018.

[8] 赵旦峰, 李文意, 杨建华, 等. 分块并行 Turbo 码译码算法的研究[J]. 哈尔滨工程大学学报, 2004, 25(2): 209-212.

[9] 樊宏伦,袁悦. 低码率Turbo最优编码设计 [J]. 电子技术. 2025 (08)：83-85.

[10] 董咏潮，灵活全并行Turbo译码算法设计及实现[D].北京邮电大学，2025.

[11] Martin-Vega, Francisco Javier,et al.“Further improvements in SOVA for high-throughput parallel turbo decoding.” IEEE Communications Letters 19.1(2014):6-9.