

实验4 差错控制编码实验报告

5 实验内容

5.1 (7,4)汉明码的纠错实验

- (1) 记录不同信道误符号率 ε 下，有汉明码编译码和无汉明码编译码时的误块率和误比特率。

信道误符号率		0.001	0.005	0.01	0.05	0.1	0.2
无汉明码编译码	数据块数	1000000	200000	100000	20000	10000	10000
	误比特率	0.0010	0.0050	0.0010	0.0504	0.1004	0.2007
	误块率	0.0041	0.0197	0.0394	0.1865	0.3454	0.5954
有汉明码编译码	数据块数	1000000	200000	100000	20000	10000	10000
	误比特率	7.5×10^{-6}	2.16×10^{-4}	9.1×10^{-4}	0.0200	0.0682	0.1967
	误块率	1.9×10^{-5}	4.95×10^{-4}	0.0022	0.0458	0.1554	0.4270

表 1: 差错控制编码实验记录

- (2) 给出误码率随信道误符号率变化的曲线图。

见图 1.

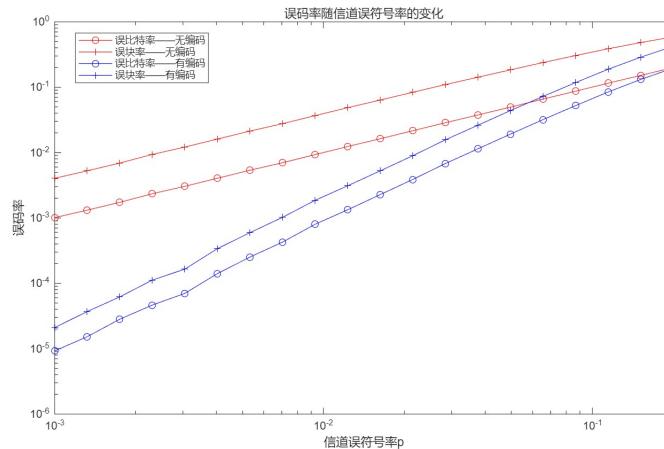


图 1: 误码率随信道误符号率变化曲线图

- (3) 分析曲线变化原因。

在低信道误符号率时，有编码的误比特率明显低于无编码的情况，这是因为汉明码能够有效纠正单个随机错误。

在高信道误符号率时，有编码的误比特率可能接近甚至高于无编码的情况，这是因为当错误位数超过1位时，汉明码可能产生错误的纠错，导致更多错误，这体现了汉明码只能纠正单个错误的局限性。

5.2 (7,4)汉明码的交织实验

(1) 调试交织函数`interleaver`和解交织函数`deinterleaver`

- (a) 给出使用[1,2,3,...,35]的整数序列时，交织前后及解交织前后的序列。

交织前序列: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35

交织后序列: 1 8 15 22 29 2 9 16 23 30 3 10 17 24 31 4 11 18 25
32 5 12 19 26 33 6 13 20 27 34 7 14 21 28 35

解交织后序列: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35

- (b) 给出使用全零的整数序列，突发错误长度为交织块行数时，解交织前后的错误图案。

解交织后序列: 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0

- (c) 给出使用全零的整数序列，突发错误长度为交织块行数的2倍时，解交织前后的错误图案。

解交织后序列: 0 0 1 1 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0

(2) 差错控制编码与交织

- (a) 给出所使用的生成矩阵 \mathbf{G} 。生成矩阵 \mathbf{G} 如下:

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

- (b) 给出原始的信息序列 x ，汉明码编码后的序 x_code ，交织后的序列 $x_interleave$ ，经过信道传输后的序列 y ，解交织后的序列 $y_deinterleave$ ，以及纠错后的序列 y_decode 。

x : 1 1 0 1 1 0 0 1 1 1 0 1 1 0 1 0 0 1 1 1

```

x_code: 1 1 0 1 1 0 0 1 0 0 1 0 0 1 1 1 0 1 1 0 0 1 0 1 0 1 0 1 0 1
1 1 0 0 1
x_interleave: 1 1 1 1 0 1 0 1 0 1 0 0 0 1 1 1 1 1 0 1 1 0 1 1 0 0
0 0 0 0 1 0 1 1
y: 1 1 1 1 0 1 0 1 0 1 0 0 0 1 1 1 1 1 0 1 0 0 1 1 0 0 0 0 0 0 1 0
1 1
y_deinterleave: 1 1 0 1 0 0 0 1 0 0 1 0 0 1 1 1 0 1 1 0 0 1 0 1 0 1
0 1 0 1 1 1 0 0 1
y_decode: 1 1 0 1 1 0 0 1 1 1 0 1 1 0 1 0 0 1 1 1

```

- (c) 分析交织的作用和效果。交织的作用是使一个突发错误产生的错误比特分散至多个块中，从而尽量使每个块中的误比特数不多于1，从而发挥汉明码的纠错作用，降低误比特率。
效果是大大降低了最终的误比特率，甚至可将误比特率清零。
- (d) 记录不同信道突发错误长度下，有无交织/解交织时的误块率和误比特率。

信道突发错误长度		3	5	10	15	20	25
无交织/解交织	数据块数	10000	10000	10000	10000	10000	10000
	误比特率	0.0443	0.0790	0.1499	0.2190	0.2909	0.3670
	误块率	0.0885	0.1551	0.2907	0.4226	0.5544	0.6996
有交织/解交织	数据块数	10000	10000	10000	10000	10000	10000
	误比特率	0	0	0.1254	0.2505	0.3473	0.4087
	误块率	0	0	0.2474	0.4960	0.6884	0.8166

表 2: 差错控制编码实验记录

6 实验总结、体会和建议

通过本次差错控制编码实验，我深入理解了汉明码的基本原理与性能特点，掌握了(7,4)汉明码的编码、译码及纠错过程。实验结果表明，在低误码率信道环境下，汉明码能够有效纠正单个随机错误，显著降低系统误比特率和误块率，体现了信道编码在提升通信可靠性方面的重要价值。然而当信道误符号率较高时，由于汉明码最小汉明距离为3，仅能纠正单错而无法应对多错，甚至可能因错误传播导致性能恶化，这让我深刻认识到编码方案的选择需与实际信道条件相匹配。在交织实验中，我进一步体会到交织技术通过时间维度上的分散处理，能够将突发错误转化为随机

错误，从而充分发挥汉明码的纠错能力，这种“时间分集”的思想为解决无线通信中的深度衰落问题提供了有效途径。整个实验过程不仅锻炼了我的MATLAB编程与仿真能力，更让我对差错控制编码的理论与实践有了系统性的认识，理解了在实际通信系统中如何通过编码与交织的联合设计来平衡可靠性、效率与复杂度，为后续深入学习现代通信技术奠定了坚实基础。