

# 面向水声网络可靠传输的 FDR 编解码算法

王丽娟<sup>1</sup>, 杜秀娟<sup>1,2</sup>, 李冲<sup>1</sup>

(1. 青海师范大学计算机学院, 青海 西宁 810008;  
2. 高原科学与可持续发展研究院, 青海 西宁 810008)

**摘要:** 通过分析 RLT 码, 针对 RLT 码译码算法中存在的缺陷, 提出一种过滤式降维 (FDR) 算法, 消除了传统译码算法在收到一定数量编码分组才开始解码的等待时间, 实现了边接收边尝试解码的快速译码方式。此外, 通过编码分组之间的异或运算, 有效增加了度为 1 的编码分组的产生概率, 不再仅依赖于从发送端获取度为 1 的编码分组, 在降低传输时延的同时通过增加度为 1 的编码分组出现的概率从而提高译码成功率。在此基础上, 提出一种与 FDR 译码算法相结合的优化度分布函数, 适当提高度为 2、度为 3、度为 4 的编码分组的比例, 从而大大增加一次降维即可得到度为 1 的编码分组的概率, 加快了译码速度。NS3 仿真结果表明, FDR 算法的解码成功率普遍高于 RLT 码。

**关键词:** 水声通信; 可靠传输; 数字喷泉码; 过滤式降维

**中图分类号:** TP393

**文献标识码:** A

**doi:** 10.11959/j.issn.1000-436x.2020058

## FDR coding and decoding algorithm for reliable transmission in underwater acoustic network

WANG Lijuan<sup>1</sup>, DU Xiujuan<sup>1,2</sup>, LI Chong<sup>1</sup>

1. Computer Department, Qinghai Normal University, Xining 810008, China  
2. Academy of Plateau Science and Sustainability, Xining 810008, China

**Abstract:** By analyzing the shortcomings of RLT coding and decoding algorithm, a filtering dimension reduction (FDR) algorithm was proposed, which eliminated the waiting time of the traditional decoding algorithm and achieves fast decoding. In addition, XOR operation between encoded packages effectively increased the number of one-degree encoded packages, and improved decoding probability while reducing transmission delay. An optimized degree distribution function for FDR decoding algorithm was proposed, which increased the proportion of two-degree, three-degree and four-degree encoded packages, further increased the probability of one-degree packet, so that speeded up the decoding progress. Simulation results with NS3 show that the decoding success probability of FDR algorithms higher than RLT algorithm.

**Key words:** underwater acoustic communication, reliable transmission, digital fountain code, filtering dimension reduction

### 1 引言

随着海洋逐渐成为人们维持可持续发展的第

二领域, 水声通信 (UAC, underwater acoustic communication) [1-5] 以其水下远距离传输的独特优势和巨大的社会效益而备受关注。陆地上的互联网

收稿日期: 2019-07-11; 修回日期: 2020-03-02

通信作者: 杜秀娟, dxj@qhnu.edu.cn

基金项目: 国家自然科学基金资助项目 (No.61962052, No.61902273); 青海省自然科学创新团队基金资助项目 (No.2020-ZJ-903); 青海省物联网重点实验室基金资助项目 (No.2020-ZJ-Y16); 河北省物联网监测中心基金资助项目 (No.3142016020); 教育部“春晖计划”基金资助项目

**Foundation Items:** The National Natural Science Foundation of China (No.61962052, No.61902273), The Innovation Team Foundation of Qinghai Office of Science and Technology (No.2020-ZJ-903), Key Laboratory of IoT of Qinghai (No.2020-ZJ-Y16), Hebei IoT Monitoring Center (No.3142016020), The Research Fund for the Chunhui Program of Ministry of Education of China

通信中, 不仅应用种类很多, 数据量还很庞大。而对于水声通信, 基本的数据传输很难保障, 但是人类赖以生存的环境与海洋息息相关, 人类不得不了解海洋以达到科学利用海洋的目的。克服水声通信中窄带宽、长时延、高误码率、多普勒频移、噪声及时变性的困难, 通过数字喷泉码提高水下数据传输的可靠性对构建高稳健性水声网络具有重要意义<sup>[6-9]</sup>。

目前, 水声网络可靠传输技术分为 3 类, 分别是基于重传、纠错、纠错和重传的可靠传输算法。数字喷泉码属于基于纠错的可靠传输算法。较早的面向水声网络可靠传输的机制有很多, 例如, Guo 等<sup>[10]</sup>将网络编码和多径路由相结合提出了网络编码机制; Liu 等<sup>[11]</sup>提出了包级的前向纠错 (FEC, forward error correction), 用于可靠传输; Xie 等<sup>[12]</sup>提出了一种分段数据可靠传输方案; Liu 等<sup>[13]</sup>提出了最初的自适应可靠传输协议 (ARTP, adaptive reliable transport protocol), 根据节点间的距离找到合适的 FEC 组合, 实现水下数据的可靠传输; Mo 等<sup>[14]</sup>提出了一种基于编码的多跳协调可靠数据传输机制。

数字喷泉码对中间编码信息的出错并不敏感, 只关注接收端最终是否能成功译码, 并且即使传输过程发生个别信息丢失也不会进行频繁的数据重传<sup>[15-16]</sup>。降低编码信息与节点、信道的相关性, 是数字喷泉码适用于水声网络的一大优势。递归 LT (RLT, recursive LT) 码<sup>[17-18]</sup>是一种基于数字喷泉码的递归 LT 码, 其采用递归思想改进了 LT 码的编码算法。RLT 解码算法依赖度为 1 的编码分组, 如果度为 1 的编码分组在信道传播过程中发生丢失或误码, 则解码极有可能会失败。如果在解码时从编码分组之间异或的角度出发, 只要生成矩阵中 2 个编码分组对应的列存在“严格短环”, 那么二者即可进行异或运算。

对此, 本文提出一种过滤式降维 (FDR, filtering dimension reduction) 译码算法, 对 RLT 码译码算法做出改进, 弥补 RLT 码的译码缺陷, 提高译码成功率。FDR 算法利用生成矩阵中构成严格短环的编码分组之间的异或操作, 产生度为 1 的编码分组或者将高度数编码分组进行“降维”处理, 使其度数降低。不会像 RLT 译码算法那样, FDR 译码算法不仅解除了对来自发送端编码器产生的度为 1 的编码分组的依赖, 还可以快速降低高度数编码分组的度数, 从而进一步降低译码复杂度。此外, 本文提出了一种与 FDR 译码算法相结合的优化度分布函数, 在保证度为 1 的编码分组数量适当的前提下,

增大了高度数编码分组的选取概率, 从而增加了一次降维即可得到度为 1 的编码分组的概率, 加快了译码进度。

本文的贡献如下。

1) 对 RLT 码中存在的短环问题进行数学分析, 在此基础上定义、分析严格短环问题, 进一步对严格短环加以利用, 提出 FDR 算法, 解决了对度为 1 的编码分组的依赖。

2) 提出一种与 FDR 算法相结合的优化度分布函数, 增加了 FDR 算法在进行解码时一次降维即可得到度为 1 的编码分组的概率。

3) 将 FDR 算法和 RLT 码在 NS3 仿真平台上进行大量测试, 经对比分析, 实验结果验证了 FDR 算法的有效性和稳健性。

## 2 相关工作

涉及本文研究内容的相关算法包括 RLT 编码算法和 RLT 解码算法<sup>[17-18]</sup>。

### 2.1 RLT 编码算法

RLT 编码器能够由原始输入分组产生无限多个编码分组序列。对于给定的  $(k, d, \Omega(d))$ , 其中  $k$  为原始分组个数,  $d$  ( $d \in \{1, 2, 3, 4, k\}$ ) 为编码分组的度值,  $\Omega(d)$  为度分布, 输入分组序列为  $\{S_1, S_2, \dots, S_k\}$ ,  $k$  个输入分组构成集合  $I$ , 编码分组序列为  $\{Y_1, Y_2, \dots, Y_j, \dots, Y_n\}$  ( $n > k$ )。编码分组与原始分组对应关系如图 1 所示, 具体产生过程如下。

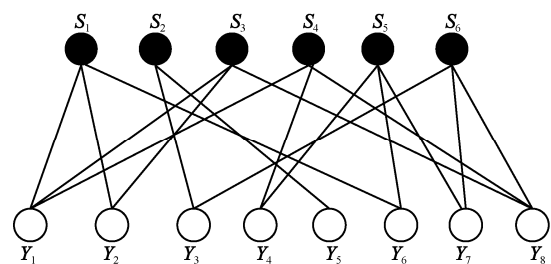


图 1 编码分组与原始分组对应关系

1) RLT 编码器对集合  $I$  中的  $k$  个输入分组先后分别执行异或操作, 产生一个度为  $k$  的编码分组, 复制得到  $\left\lceil \frac{1}{1-p_p} \right\rceil$  个度为  $k$  的编码分组, 其中  $p_p$  表示水下数据逐跳传输的分组错误率。

2) 从集合  $I$  中随机选取  $\left\lceil \frac{m}{1-p_p} \right\rceil$  个不同的分组,

构成集合 $U_1$ ，产生 $\left\lfloor \frac{m}{1-p_p} \right\rfloor$ 个度为1的编码分组，

其中 $m \left( 1 \leq m \leq \max \left( \left\lfloor \frac{k}{4} \right\rfloor, 1 \right) \right)$ 表示期望接收到的度为1的数据分组个数。

3) 令 $U_2 = I - U_1$ ，从集合 $U_2$ 中随机选取 $\left\lfloor \frac{k}{2(1-p_p)} \right\rfloor$ 个输入分组，分别与从 $U_1$ 中随机选取的一个分组进行异或操作，从而产生 $\left\lfloor \frac{k}{2(1-p_p)} \right\rfloor$ 个度为2的编码分组。

4) 令 $U_3 = I - U_1 - U_2$ ，若 $|U_3| < \left\lfloor \frac{k}{6(1-p_p)} \right\rfloor$ ，则从 $U_3$ 中随机选取 $|U_3|$ 个输入分组，从集合 $I$ 中随机选取 $\left\lfloor \frac{k}{6(1-p_p)} \right\rfloor - |U_3|$ 个输入分组；否则从 $U_3$ 中随机选取 $\left\lfloor \frac{k}{6(1-p_p)} \right\rfloor$ 个输入分组，分别与从 $U_1$ 、 $U_2$ 各自随机选取的一个分组进行异或操作，从而产生 $\left\lfloor \frac{k}{6(1-p_p)} \right\rfloor$ 个度为3的编码分组。

5) 令 $U_4 = I - U_1 - U_2 - U_3$ ，若 $|U_4| < \left\lfloor \frac{\xi + \frac{k}{3} - m - 1}{1-p_p} \right\rfloor$ ，则从 $U_4$ 中随机选取 $|U_4|$ 个输入分组，从集合 $I$ 中随机选取 $\left\lfloor \frac{\xi + \frac{k}{3} - m - 1}{1-p_p} \right\rfloor - |U_4|$ 个输入分组；否则从 $U_4$ 中随机选取 $\left\lfloor \frac{\xi + \frac{k}{3} - m - 1}{1-p_p} \right\rfloor$ 个输入分组，分别与从 $U_1$ 、 $U_2$ 、 $U_3$ 各自随机选取的一个分组进行异或操作，从而产生 $\left\lfloor \frac{\xi + \frac{k}{3} - m - 1}{1-p_p} \right\rfloor$ 个度为4的编码分组。

### 2.2 RLT 解码算法

当编码分组通过删除信道传输时，要么被接收节点正确接收，要么产生分组丢失。RLT 解码器试

图从接收的编码分组中恢复原始输入分组，RLT 解码过程如下。

- 1) 首先找到仅与一个输入分组 $X_i$ 相连的编码分组 $Y_j$ ，若找不到，终止译码。
- 2) 令 $X_i = Y_j$ 。
- 3) 在图1中，对每一个连接到 $X_i$ 的编码分组 $Y_j$ 进行异或操作，即 $Y_m = Y_m \oplus X_i$ 。
- 4) 删除与 $X_i$ 相连的所有边。
- 5) 继续从1)执行，直至解码成功。

## 3 RLT 码存在的问题

### 3.1 短环问题

停止集 $\mathcal{S}$ <sup>[19]</sup>是满足如下条件的变量节点的集合：任意与 $\mathcal{S}$ 所包含变量节点相连的校验节点都至少2次连接到 $\mathcal{S}$ 。停止集包含的变量节点的数量称为停止集的尺度，停止集内的变量节点构成的生成矩阵中存在短环。生成矩阵 $\mathbf{G}_{kn}$ 是一个 $k \times n$ 阶的二进制矩阵， $\mathbf{G}_{kn}$ 含有 $n$ 个列向量，记作 $\mathbf{G}_{kn} = [\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3, \dots]$ ，列向量记作 $\mathbf{Y}_m = [Y_{1m}, Y_{2m}, \dots, Y_{km}]^T$ ， $m = 1, 2, 3, \dots, n$ ，其元素的取值为0和1。 $\mathbf{G}_{kn}$ 的示例为

$$\begin{matrix}
 \mathbf{S}_1 & \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 1 & \dots & 0 & 1 \end{bmatrix} \\
 \mathbf{S}_2 & \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 & 0 & \dots & 1 & 1 \end{bmatrix} \\
 \mathbf{S}_3 & \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 1 & \dots & 0 & 0 \end{bmatrix} \\
 \mathbf{S}_4 & \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & \dots & 0 & 1 \end{bmatrix} \\
 \mathbf{S}_5 & \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 & 0 & \dots & 1 & 1 \end{bmatrix} \\
 \vdots & \vdots \\
 \mathbf{S}_{k-1} & \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 1 & \dots & 1 & 0 \end{bmatrix} \\
 \mathbf{S}_k & \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 1 & 0 & \dots & 0 & 0 \end{bmatrix} \\
 & \mathbf{Y}_1 \quad \mathbf{Y}_2 \quad \mathbf{Y}_3 \quad \mathbf{Y}_4 \quad \mathbf{Y}_5 \quad \mathbf{Y}_6 \quad \mathbf{Y}_7 \quad \dots \quad \mathbf{Y}_{n-1} \quad \mathbf{Y}_n
 \end{matrix}$$

每个列向量对应一个由 $k$ 个输入分组（对应矩阵中 $\mathbf{S}_1 \dots \mathbf{S}_k$ ）中的某几个分组异或而得到的编码输出分组 $\mathbf{Y}_m$ ，如果向量元素取0表示该输入分组不参与此编码分组的生成，如果向量元素取1表示该输入分组参与此编码分组的生成， $\mathbf{Y}_m$ 如式(1)所示。

$$\mathbf{Y}_m = \sum_{i=1}^k x_i \mathbf{Y}_{im} \tag{1}$$

其中， $x_i$ 表示向量元素取值。下面给出短环的定义及性质。

**定义1** 短环。在生成矩阵中，如果有这样两列，它们在相同位置上的两行（或两行以上）均为1，这些由1构成的行形成一个闭合的环，称为短环。

如果满足短环定义的行有两行，那么这两行构

成的短环为4元环;如果满足短环定义的行有三行,那么这三行构成的短环为6元环,以此类推,假设满足短环定义的行有 $k'(2 \leq k' < k)$ 个,它们构成的短环为 $2k'$ 元环。以图2所示的RLT译码过程中出现终止现象为例解释短环。

图2(a)所示原始分组和编码分组的对应关系为 $Y_1=S_2, Y_2=S_2 \oplus S_3 \oplus S_4, Y_3=S_1 \oplus S_2 \oplus S_3, Y_4=S_1 \oplus S_2 \oplus S_3 \oplus S_4$ 。根据RLT码译码过程,首先找到度为1的编码分组 $Y_1$ ,此时可直接译出 $S_2$ 并将 $Y_1$ 和 $S_2$ 之间的连线去掉,如图2(b)所示;下一步,将所有与 $S_2$ 相连的编码分组 $\{Y_2, Y_3, Y_4\}$ 分别与 $S_2$ 进行异或运算,同时更新 $Y_2, Y_3, Y_4$ 的值为异或运算结果,并且将它们之间的连线删除,如图2(c)所示。此时,剩余的编码分组 $Y_2, Y_3, Y_4$ 的度值 $d$ 分别为 $d(Y_2)=2, d(Y_3)=2, d(Y_4)=3$ 。根据LT码译码规则,需要找出度为1的编码分组,但是3个编码分组的度值均大于或等于2,所以译码被迫终止。此时, $Y_2, Y_3, Y_4$ 对应的生成矩阵如图3所示。

图3的生成矩阵中包含2个环长为4的短环: $Y_3, Y_4$ 对应的两列在第一行和第三行都是1,所以第一行和第三行构成4元短环;同样地, $Y_2, Y_4$ 所对应的两列在第三行和第四行都是1,所以这两行也构成了4元短环,如图3中虚线所示。以4元环为例,假设RLT码参数为 $(n, k, \Omega)$ ,其中 $k$ 为原始分组个数, $n$ 为编码分组个数,在 $n \times k$ 阶的生成矩阵 $G$ 中重量为 $i$ 的列向量在全体列向量中的比例是 $\Omega_i$ ,也就是说,度为 $i$ 的输出符号(即编码分组)在全体输出符号中的比例为 $\Omega_i^{[19]}$ 。

那么,生成矩阵 $G$ 中某个重量为 $i$ 的列向量的生成概率为<sup>[19]</sup>

$$P_i = \frac{\Omega_i}{\binom{k}{i}} \quad (2)$$

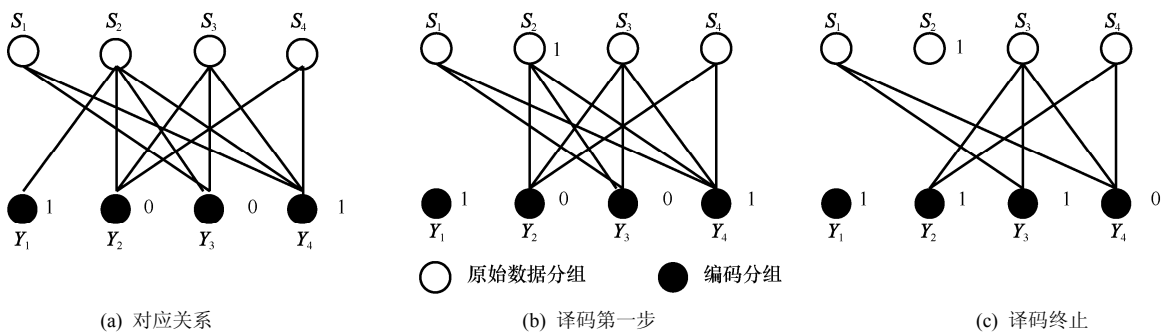


图2 RLT码译码过程中出现终止现象

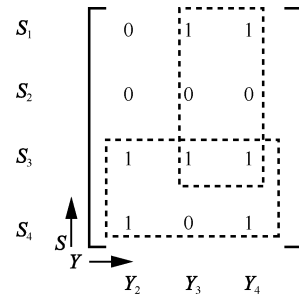


图3  $Y_2, Y_3, Y_4$ 对应的生成矩阵

生成矩阵 $G$ 中某个重量为 $j(j > 1)$ 的列向量与该重量为 $i(i > 1)$ 的列向量构成4元短环的概率为

$$P_{-4}(i, j) = \Omega_i \Omega_j \frac{\binom{k-2}{j-2}}{\binom{k}{k-j} \binom{k}{j}} \quad (3)$$

### 3.2 严格短环

**定义2** 严格短环。在生成矩阵中,如果有这样两列,它们在相同位置上的两行(或两行以上)均为1,并且其中一列除了这两行(或多行)其余位置的行上全是0,这些由1构成的行形成一个闭合的环,称为严格短环。假设满足严格短环定义的行有 $k'(2 \leq k' < k)$ 个,则它们构成的短环为严格 $(2k')$ 元环。

生成矩阵 $G$ 中重量为 $j(j > 2)$ 的列和某一重量为2的列构成严格4元短环的概率为

$$P_{\text{strict}(r_{-4})}(2, j) = \Omega_2 \Omega_j \frac{\binom{k-2}{j-2}}{\binom{k}{k-j} \binom{k}{2}} \quad (4)$$

根据式(4)可近似地推导出重量为 $j$ 的列和某一重量为 $m(m < j)$ 的列构成严格 $2m$ 元短环的概率为

$$P_{\text{strict}(r_{-2m})}(m, j) = \Omega_m \Omega_j \frac{\binom{k-m}{j-m}}{\binom{k}{k-j} \binom{k}{m}} \quad (5)$$

则对于  $k \times n$  阶生成矩阵  $G$  存在严格 4 元短环的概率为

$$P_{\text{strict}(r_{-4})} = \binom{n}{2} \sum_{j=2}^{\Omega} \Omega_2 \Omega_j \frac{\binom{k-2}{j-2}}{\binom{k}{k-j} \binom{k}{2}} \quad (6)$$

根据式(6)可近似地推导出生成矩阵  $G$  存在严格  $2m$  元短环的概率为

$$P_{\text{strict}(r_{-2m})} = \binom{n}{2} \left[ \Omega_2 \Omega_3 \frac{\binom{k-2}{j-2}}{\binom{k}{k-3} \binom{k}{2}} + \dots + \Omega_2 \Omega_k \cdot \frac{\binom{k-2}{j-2}}{\binom{k}{k} \binom{k}{2}} + \Omega_3 \Omega_4 \frac{\binom{k-3}{j-3}}{\binom{k}{k-4} \binom{k}{3}} + \dots + \Omega_3 \Omega_k \frac{\binom{k-3}{j-3}}{\binom{k}{k} \binom{k}{3}} + \dots + \Omega_{k-1} \Omega_k \cdot \frac{\binom{k-(k-1)}{k-(k-1)}}{\binom{k}{k} \binom{k}{k-1}} \right] = \binom{n}{2} \sum_{m=2}^{\Omega-1} \sum_{j=3}^{\Omega} \Omega_m \Omega_j \frac{\binom{k-m}{j-m}}{\binom{k}{k-j} \binom{k}{m}} \quad (7)$$

其中,  $m < j$ 。

## 4 FDR 算法与优化度分布函数

### 4.1 译码思想

在第3节短环问题的分析中, 当出现图2所示的RLT码译码终止现象时, 3个编码分组  $Y_2$ 、 $Y_3$ 、 $Y_4$  构成的生成矩阵中包含了2个4元严格短环。接下来, 从参与编码的原始分组集合的角度进行分析。编码分组  $Y_2$  对应的参与编码的原始分组的集合为  $\{S_3, S_4\}$ ,  $Y_3$  对应的参与编码的原始分组的集合为  $\{S_1, S_3\}$ ,  $Y_4$  对应的参与编码的原始分组的集合为  $\{S_1, S_3, S_4\}$ 。这3个集合存在明显的包含关系, 即  $\{S_3, S_4\} \subseteq \{S_1, S_3, S_4\}$ ,  $\{S_1, S_3\} \subseteq \{S_1, S_3, S_4\}$ 。将  $Y_2$  和  $Y_4$  进行异或运算可得到  $S_1 = Y_2 \oplus Y_4$ , 将  $Y_3$  和  $Y_4$  进行异或运算可得到  $S_4 = Y_3 \oplus Y_4$ , 将  $S_1$  和  $Y_3$  进行异或运算可得到  $S_3 = S_1 \oplus Y_3$ 。原本使用BP译码算法只能依赖度为1的  $Y_1$  译出  $S_2$ ,  $S_2$  的译码运算量

$C_{S_2} = 1$ , 而  $S_1, S_3, S_4$  根本无法解出, 可以认为, 获得  $S_1, S_3, S_4$  的译码运算量  $C_{S_1}, C_{S_3}, C_{S_4}$  接近于无穷大, 即  $C_{S_1}, C_{S_3}, C_{S_4} \rightarrow \infty$ , 继而得到所有原始分组的译码运算量为  $C_{\text{all}} = C_{S_1} + C_{S_2} + C_{S_3} + C_{S_4} \rightarrow \infty$ 。但是通过编码分组之间的异或运算, 即可解出3个原始分组  $S_1, S_3, S_4$ 。 $S_1, S_2, S_3, S_4$  的译码运算量分别为  $C'_{S_1} = 1, C'_{S_2} = 1, C'_{S_3} = 2, C'_{S_4} = 1$ , 那么成功解码所有原始分组的译码运算量为  $C'_{\text{all}} = C'_{S_1} + C'_{S_2} + C'_{S_3} + C'_{S_4} = 5 \ll C_{\text{all}}$ 。

表面上看, 严格短环是一种编码时的浪费, 对传统的依赖度为1的编码分组的解码方式并无价值, 但是把译码方式进行一定改变, 就可以对严格短环加以利用。严格短环的贡献将不可忽视, 甚至可能成为解码剩余的未成功的原始分组的关键步骤。因此, 有结论1成立。

**结论1** 生成矩阵中构成短环的两列只要度  $z$  值不同, 并且度值较小的那一列除了构成短环的行存在1, 其他行不存在1时, 它们所对应的编码分组之间即可进行异或运算。二者异或产生的新编码分组(称为二次编码分组)的度值必定小于二者之一, 甚至比二者度值都小。如果二者度值相差1, 它们异或直接得到一个度为1的分组; 如果二者度值相差大于1, 经过异或之后, 度值较大的那一列的度值可降低为二者度值之差。

LT码译码过程要在接收到一定数量编码分组后进行; RLT码在此做出改进, 数据编码结束后首先发送度为1的编码分组, RLT码的译码过程在接收到编码分组之时即刻启动, 但这2种译码方式终究都是依赖度为1的编码分组启动译码。而采用FDR算法的译码器, 在收到第一个编码分组(无论度值是否为1)后可以开始译码。仍以图2为例, 如果采用FDR译码算法的接收端收到的第一个编码分组是  $Y_4$ 、第二个编码分组是  $Y_3$  时, 接收端对比参与二者编码的原始分组集合, 2个集合存在真包含关系, 接收端便可启动译码过程, 对2个编码分组进行异或运算。因此, 有结论2成立。

**结论2** 接收端在收到第一个编码分组时就可以启动译码过程, 只要其对应的参与编码的原始分组集合与后续收到的编码分组对应的参与编码的原始分组集合之间存在真包含关系, 就不必等待度为1的编码分组, 这在一定程度上缩短了译码时间。

基于以上2个结论, 本节提出一种FDR译码

算法, 接下来主要介绍译码器的设计和译码流程。首先给出 FDR 译码算法中的几个参数。

**定义 3**  $k$  个输入符号向量为  $\mathbf{S} = \{\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_k\}$ , 对  $k$  个输入符号进行编码产生的  $n$  个编码符号向量为  $\mathbf{Y} = \{\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_i, \dots, \mathbf{Y}_n\}$ , 其中, 编码分组  $\mathbf{Y}_i$  的度值表示为  $d(\mathbf{Y}_i)$ 。FDR 算法将编码分组分为 2 种类型, 除了由  $k$  个原始分组经发送端编码器产生的  $n$  个编码分组  $\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_n$  之外, 还包括编码分组在接收端译码器内与其他编码分组异或产生的二次编码分组。为了区分, 二次编码分组用  $\mathbf{Y}_{\text{sec}}$  表示, 那么二次编码分组的度值表示为  $d(\mathbf{Y}_{\text{sec}})$ 。一个编码分组 (无论  $\mathbf{Y}_1, \mathbf{Y}_2, \dots, \mathbf{Y}_n$  中的某一个  $\mathbf{Y}_i$  还是某一个二次编码分组  $\mathbf{Y}_{\text{sec}}$ ) 对应参与其编码原始分组的 ID 集合为  $T$ 。

### 4.2 译码器的设计

译码器根据编码分组的度值范围采用分层设计思想。在本文 4.4 节提出的度分布函数中, 编码分组的度值  $d(d \in \mathbb{Z})$  有 5 种, 分别为  $d=1$ 、 $d=2$ 、 $d=3$ 、 $d=4$ 、 $d=k$ 。译码器随之设计为 5 层, 即  $l_1$ 、 $l_2$ 、 $l_3$ 、 $l_4$ 、 $l_k$ , 如图 4 所示。译码器的每层分别存放的是相应度值的编码分组, 在这里, 层内的编码分组既包括来自接收端的编码分组, 也包括来自发送端的编码分组进入译码器后在各层之间流动时与层内已存在的 (先进入译码器的) 编码分组异或产生的二次编码分组。比如,  $l_2$  存放了所有度为 2 的编码分组, 这些编码分组当中可能有接收到来自发送端的度为 2 的编码分组, 还可能有一个度为 3 的编码分组与一个度为 1 的编码分组在  $l_1$  层异或产生的度为 2 的二次编码分组。这里还需说明的是,  $l_k$  层存放度值范围为  $d \in (4, k]$  的编码分组, 度为  $k$  的编码分组可以和任何一个编码分组进行异或, 那么一个度为  $k$  (假设  $k > 8$ ) 的编码分组和一个度为  $t(t=1, 2, 3, 4)$  的编码分组异或产生的二次编码分组的度值为  $k-t$  且  $k-t > 4$ , 因此将度值大于 4 且小于  $k$  的二次编码分组放入  $l_k$  层。以度为 4 的编码分组在  $l_k$  层发生异或运算为例, 产生的二次编码分组  $\mathbf{Y}_{\text{sec}}$  的度为  $d=k-4=6$ , 所以将二次编码分组  $\mathbf{Y}_{\text{sec}}$  放在  $l_k$  层。编码分组在译码器内的存在形式是一对 key-value, key 表示该编码分组对应的参与构成其编码原始分组的 ID 集合  $T$ , value 表示该编码分组。例如, 编码分组  $\mathbf{Y}_i = \mathbf{S}_1 \oplus \mathbf{S}_2 \oplus \mathbf{S}_3$ ,  $\mathbf{S}_1$ 、 $\mathbf{S}_2$ 、 $\mathbf{S}_3$  对应的 ID 分别为 0、1、2, 即  $T_{\mathbf{Y}_i} = \{0, 1, 2\}$ , 那么  $\mathbf{Y}_i$  在  $l_3$  层的存在形式为  $\{0, 1, 2\} - \mathbf{Y}_i$ 。编码分组进入译码器

后向下流动, 依次经过译码器的各层, 当到达某一层时, 如果该编码分组的  $T$  和当前层内某个编码分组的  $T$  存在真包含关系, 换句话说, 度值小的编码分组里的所有原始数据分组同样参与了度值较大的编码分组的异或运算 (此为异或条件), 那么二者异或运算之后, 度值较大的编码分组被更新为二次编码分组, 其度值被降低。此过程相当于对度值大的编码分组进行过滤、降维, 降低高度数编码分组的度值, 在一定程度上加快了译码进度, 度值较大或者度值很小的编码分组在某层与其他编码分组产生异或运算的概率更大。

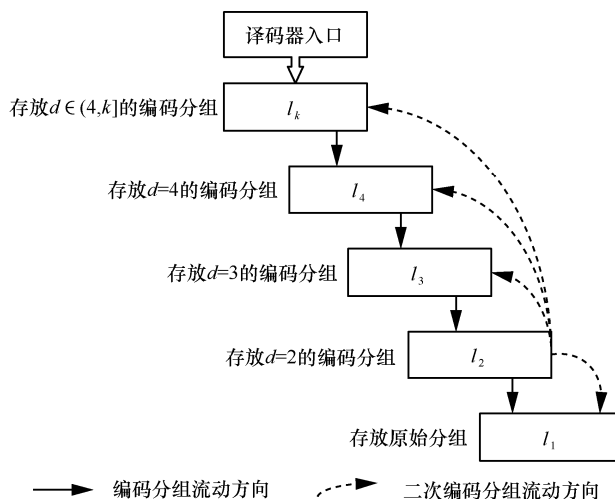


图 4 译码器层次设计

理论上, 译码器的  $l_2$ 、 $l_3$ 、 $l_4$ 、 $l_k$  层与它下面除  $l_1$  层外所有层内的每一个 key 不存在真包含关系。FDR 译码器从收到第一个编码分组开始, 一直处在解码状态, 直至  $l_1$  层包含整个向量  $\mathbf{S}$ , 视为译码成功。

### 4.3 FDR 译码流程

FDR 算法规定发送端将数据编码完成后, 先发送  $d=k$  的编码分组  $\mathbf{Y}_n$ , 再依次发送  $d=4$ 、 $d=3$ 、 $d=2$ 、 $d=1$  的编码分组。具体流程如算法 1 所示。

**算法 1** 面向水声网络可靠传输的 FDR 编解码算法

输入 编码分组序列  $\mathbf{Y} = \{\mathbf{Y}_n, \mathbf{Y}_{n-1}, \dots, \mathbf{Y}_2, \mathbf{Y}_1\}$

输出 原始分组  $\mathbf{S} = \{\mathbf{S}_k, \mathbf{S}_{k-1}, \dots, \mathbf{S}_2, \mathbf{S}_1\}$

1) if  $l_1$  层的向量  $\mathbf{S}$  未包含所有原始分组 then

2) if  $i = n$  then

3) 收到了第一个编码分组  $\mathbf{Y}_n$ , 因为  $d(\mathbf{Y}_n) = k$ , 将其放入  $l_k$  层;

- 4) else
- 5) 将后续收到的编码分组  $Y_i$  从  $l_k$  层开始, 与层内的编码分组逐一对比, 判断是否满足异或条件;
- 6) 若  $Y_i$  与某编码分组  $Y_j$  满足异或条件且  $d(Y_i) < d(Y_j)$ , 则  $Y_{\text{sec}} = Y_i \oplus Y_j$ , 然后将度值较小的  $Y_i$  放入其对应层; 度值较大的  $Y_j$  直接丢弃, 不再保存; 计算产生的二次编码分组的度值  $d(Y_{\text{sec}})$ , 将其放入相应层; 如果该编码分组直到  $l_1$  层始终未满足异或条件, 根据度值将其放入对应层;
- 7) end if
- 8) else
- 9) 原始分组未恢复, 将  $l_1$  层的原始分组逐层与  $l_2$ 、 $l_3$ 、 $l_4$  层的编码分组进行异或, 直至向量  $S$  包含所有原始分组;
- 10) end if

#### 4.4 与 FDR 相结合的优化度分布函数

对于给定的 RLT 码编码参数  $(n, k)$ , 其中  $k$  为原始分组数量,  $n$  为对  $k$  个原始分组进行编码后得到的编码分组数量。编码分组的入度定义为参与其构成的原始分组的个数。

设每个编码分组的平均入度值为  $D_{\text{per}}$ , 所有编码分组的总入度值为

$$\text{inDgree\_all} = nD_{\text{per}} \quad (8)$$

在形成一个编码分组时, 某个原始分组第一次被选到的概率为  $\frac{1}{k}$ , 第一次未被选到的概率为  $1 - \frac{1}{k}$ , 那么某个原始分组在整个编码过程中始终未被选取的概率  $P_{\text{miss}}$  为

$$P_{\text{miss}} = \left[ \left(1 - \frac{1}{k}\right) \left(1 - \frac{1}{k-1}\right) \cdots \left(1 - \frac{1}{k - (D_{\text{per}} - 1)}\right) \right] \quad (9)$$

不理想覆盖问题使未被覆盖 (即未参与任何编码分组的编码) 的原始分组无法译出, 是直接导致译码失败的主要原因之一, 所以在度分布的设计中必须考虑这一点。

数字喷泉码的度分布函数影响编码复杂度, 关系到编码效率。合理的度分布应该在保证度为 1 的编码分组的数量适当的前提下增大度值较大的编码分组的选取概率, 这样使由度分布产生的编码分组平均度值较小, 同时兼顾不理想覆盖问题, 保证对所有原始分组的良好覆盖。

给定  $k$  个原始数据分组, 度分布设计为

$$\Omega(d) = \begin{cases} \frac{\ln k}{\sqrt{k}(d+1)}, & d=1 \\ \frac{\rho(d) + \tau(d)}{\sum_d (\rho(d) + \tau(d))}, & d=2,3,4 \\ \frac{1}{k}, & d=k \end{cases} \quad (10)$$

其中,  $\rho(d)$  和  $\tau(d)$  的计算式分别为

$$\rho(d) = \frac{1}{2^d}, \quad d=2,3,4 \quad (11)$$

$$\tau(d) = \frac{1}{d(d+1)}, \quad d=2,3,4 \quad (12)$$

当  $d=1$  时, 编码分组的数量与  $k$  相关, 原始分组数量  $k$  越大, 度为 1 的编码分组数量越多, 越有利于加快解码速度; 当  $d=k$  时, 编码分组数量为 1, 保证所有原始分组都参与到该分组的编码中, 避免了不理想覆盖问题。 $d=2$ 、 $d=3$ 、 $d=4$  的设计从理想孤波分布出发, 将度为 2、度为 3、度为 4 的选取概率设置为小于 1 的数, 从而将度分布的优化转化成常数的设计问题。

由式(10)可得, 度分布的期望值为

$$\begin{aligned} \lambda = X(d) &= \sum_{d=1}^k (d\Omega(d)) = \frac{\ln k}{\sqrt{k}(1+1)} + \\ &2 \times \frac{\rho(2) + \tau(2)}{\sum_d (\rho(d) + \tau(d))} + 3 \times \frac{\rho(3) + \tau(3)}{\sum_d (\rho(d) + \tau(d))} + \\ &4 \times \frac{\rho(4) + \tau(4)}{\sum_d (\rho(d) + \tau(d))} + k \times \frac{1}{k} = \frac{\ln k}{2\sqrt{k}} + 3.27 \end{aligned}$$

通常,  $\frac{\ln k}{2\sqrt{k}}$  是一个不大于 0.3 的数, 所以

$\lambda \approx 3.57$ , FDR 的译码复杂度为 3.57, 与原始分组数量  $k$  无关。几种度分布的编解码复杂度比较如表 1 所示。

表 1 几种度分布的编解码复杂度比较

度分布	编解码复杂度
理想孤波分布	$k \ln_e k$
FDR 码	3.57
RLT 码	3.7
稳健式孤波分布	$\sqrt{k} \ln_e k$
二进制指数分布	2

### 4.5 编解码统计分析

将 4.4 节度分布函数纳入递归编码思想中，并采用 FDR 解码算法进行解码，本节从统计角度出发对解码成功概率进行如下分析。

对于完全随机的传统数字喷泉码，当节点接收到  $\sqrt{k} \ln_e^k$  个编码分组时，能成功解码  $k$  个编码分组的概率即为  $k \times n$  阶生成矩阵的满秩概率，如式(13)所示

$$\psi_k(n) = \prod_{i=0}^{k-1} (1 - 2^{-(i-n)}) \quad (13)$$

设水声信道的纠删概率为  $p_p$ ，编码分组在信道上的传输符合二项分布，由概率质量函数  $B(N; n, p_p)$  可知，当发送  $N$  个编码分组时，成功接收到  $n$  个编码分组的概率为

$$B(N; n, 1 - p_p) = \binom{N}{n} (1 - p_p)^n p_p^{N-n} \quad (14)$$

在完全随机数字喷泉码中，发送节点发送  $N$  个编码分组，接收节点成功解码的概率为

$$\varphi_{N,k}(p_p) = \sum_{n=k}^N B(N; n, 1 - p_p) \left( \prod_{i=0}^{k-1} (1 - 2^{-(i-n)}) \right) \quad (15)$$

由于采用 FDR 算法， $k \times n$  阶生成矩阵中满足严格短环的列向量在停止集中仍可继续被用来解码。发送同等数量的编码分组，采用 FDR 算法的通信双方中的接收节点成功解码的概率要比完全随机的数字喷泉码的解码概率高。当发送节点发送  $N$  ( $N$  由式(16)给出) 个编码分组时，接收节点能够恢复  $k$  个原始分组的概率由式(17)给出。

$$N = \sum k\Omega(d) \quad (16)$$

$$\varphi_{\omega,k}(p_p) = \sum_{\omega=1}^O B(\sum k\Omega(d); k\omega, 1 - p_p) (\psi_k(k\omega)) > \sum_{\omega=1}^O B(\sum k\Omega(d); k\omega, 1 - p_p) \left( \prod_{i=0}^{k-1} (1 - 2^{-(i-k\omega)}) \right) \quad (17)$$

其中， $O = \sum \Omega(d)$  且  $O > 1$ 。

## 5 仿真评估

### 5.1 度分布仿真实验

本节采用 Matlab 工具，对理想孤波分布 (ISD, ideal soliton distribution)、稳健式孤波分布 (RSD, robust soliton distribution)、二进制指数分布 (BED, binary exponential distribution) 和本文优化度分布进行仿真对比。

图 5 给出当  $k=10$  时，度值从 1~10 的编码分组

概率分布。图 6 给出当  $k=20$  时，度值从 1~20 的编码分组概率分布。图 7 给出当  $k=40$  时，度值从 1~40 的编码分组概率分布。

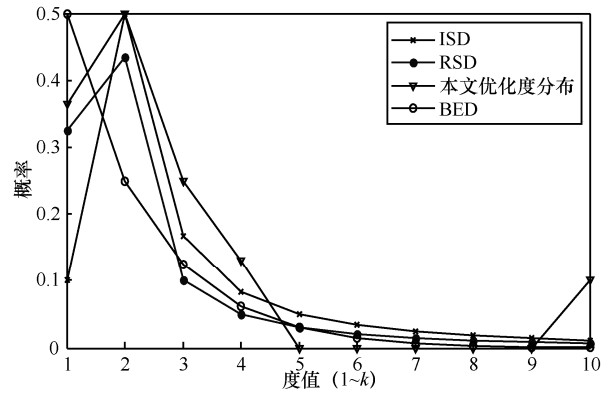


图 5  $k=10$  时的编码分组概率分布

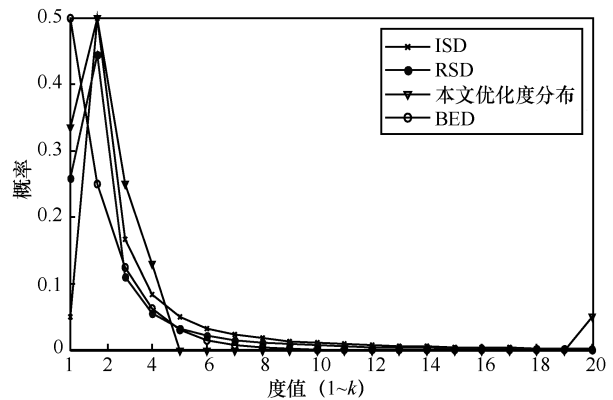


图 6  $k=20$  时的编码分组概率分布

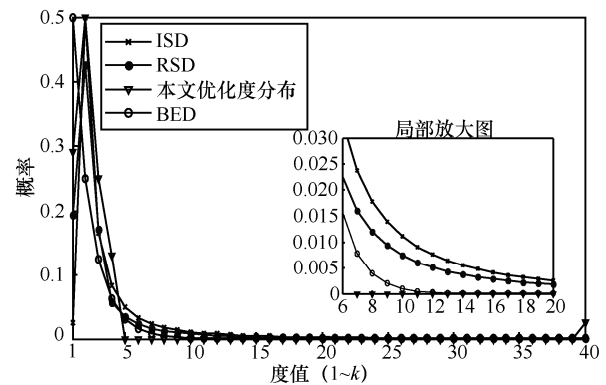


图 7  $k=40$  时的编码分组概率分布

从图 5~图 7 中可以看出，采用本文优化度分布的编码分组的平均度值较低，编码分组的数量与  $k$  相关，原始分组数量  $k$  越大，度为 1 的编码分组数量越多。度为 1、2、3、4 的编码分组的概率较其他度分布稍大；度大于 4 且小于  $k$  的编码分组数量为 0。在原始分组数量较多的情况下，这样的度分布既可以控制

解码不至于太复杂,又可以保证较高的解码成功概率,且无论 $k$ 值如何, $d=k$ 的编码分组一直存在,这保证了所有原始分组都参与编码。

## 5.2 FDR 算法仿真实验

### 1) 仿真场景

仿真实验中布置了7个水下节点,其拓扑结构如图8所示。其中6个节点构成一个二维的正六边形且分别位于6个顶点,是数据源节点;剩余一个节点是网络中的sink节点,位于正六边形的中心。数据的流动方向是源节点到sink节点,且它们都是单跳通信,源/目的节点间距和传输半径均为 $r=1\ 500\text{ m}$ 。为了排除数据分组碰撞给数据恢复带来的影响,实验中设置这6个源节点不能同时向sink节点发送数据,这可通过控制各源节点发送分组的时间来实现。

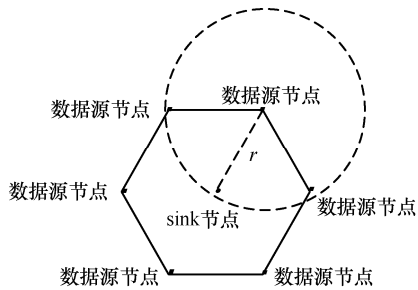


图8 正六边形网络拓扑

### 2) 仿真参数

实验中采用的协议架构为micro-ANP模型,节点在应用层产生的源数据被分为多个大小为60个数据分组的数据块,每个数据分组分为帧头、负载和帧尾3个部分。数据分组格式如表2所示。其中负载部分大小为200 B,尾部为FCS校验位。

表2 数据分组格式

比特数/bit	字段	位置
8	源节点层级(上一跳节点)	帧头部
8	源节点ID(上一跳节点)	帧头部
8	目的节点ID(下一跳节点)	帧头部
2	帧类型 00:data 01:ACK	帧头部
6	帧序列号	帧头部
1	是否立即确认	帧头部
23	原始分组ID	帧头部
变长	数据	负载
16	FCS	帧尾

仿真实验的参数设置对实验结果影响较大,调整系统参数以较准确地模拟水下通信环境,如节点的发

送增益。实验中,通过改变2个节点间的距离和发送增益,测得对应距离下的发送增益值,如表3所示。拓扑中节点间距为1 500 m,发送增益为-37 dB,节点的接收功率和休眠状态下的功率分别是0.75 W和10 mW,声波可用的频带宽度是10 kHz。

表3 传输范围与发送增益

传输范围/m	发送增益/dB
500	-49.6
1 000	-42
1 500	-37
2 000	-32
2 500	-28

此外,依据水下声信号衰减模型,仿真实验模拟了信号在水声信道传输时的损耗。实验采用Micro-ANP通信架构,信号到达接收节点的物理层,接收节点计算发送增益与路径损耗的差值得出接收增益,通过判断接收增益的阈值,物理层决定这个分组是否继续向上层递交。衰减模型具体计算式为

$$A(x) = x^k a^x \quad (18)$$

其中, $x$ 表示收发节点间距离, $a$ 的计算式为

$$a = 10^{\frac{\alpha(f)}{10}} \quad (19)$$

$$\alpha(f) = 0.11 \times \frac{f^2}{1+f^2} + 44 \times \frac{f^2}{4100+f^2} + 2.75 \times 10^{-4} f^2 + 0.003 \quad (20)$$

通常,能量扩散因子采用柱体扩散因子 $k=1.5$ ,中心频率 $f=22\text{ kHz}$ 。节点初始能量、数据块大小、仿真时间等实验参数设置如表4所示。

表4 实验参数设置

参数	取值
节点初始能量/J	1 000
数据块大小/B	2 975~8 955
分组负载长度/B	200
仿真时间/s	6 000
路由协议	LB-AGR
MAC协议	RCHF

### 3) 仿真结果

此处定义单跳的成功解码概率,实验中可测得的数据是源节点向sink节点发送数据的总次数 $N_{\text{Total-trans}}$ 及经历二次编码后成功的次数 $N_{\text{retrans}}$ ,二者之差即为单次成功传输的次数 $N_{\text{one-time-succ}}$ , $N_{\text{one-time-succ}}$

和  $N_{Total-trans}$  的比值即为一次发送数据中接收端成功解码的概率，如式(21)所示。

$$P_{Dec-succ-hop} = \frac{N_{Total-trans} - N_{retrans}}{N_{Total-trans}} = \frac{N_{one-time-succ}}{N_{Total-trans}} \quad (21)$$

FDR 译码算法和 RLT 译码算法在单次传输解码成功概率方面的仿真结果如图 9 所示，其中横轴表示单次发送的编码分组数量，纵轴表示单次传输成功解码概率。从图 9 可以看出，使用 FDR 译码算法的成功解码概率普遍高于 RLT 译码算法。当编码分组数量较少时，比如  $n=20$ ，2 个译码算法的解码成功概率相差无几，RLT 译码算法达到 92%，FDR 译码算法可达 93%。当  $n=25$  时，FDR 译码算法出现突变情况，解码成功概率为 90%，略低于 RLT 译码算法的 92%。当  $n=30$  时，FDR 译码算法性能明显优于 RLT 译码算法性能，二者分别为 95%和 89%。当  $n>40$  时，RLT 译码算法的成功概率基本保持在 86%，而 FDR 译码算法的成功概率保持在 89%左右。综合来看，FRD 译码算法的性能较 RLT 译码算法更优。

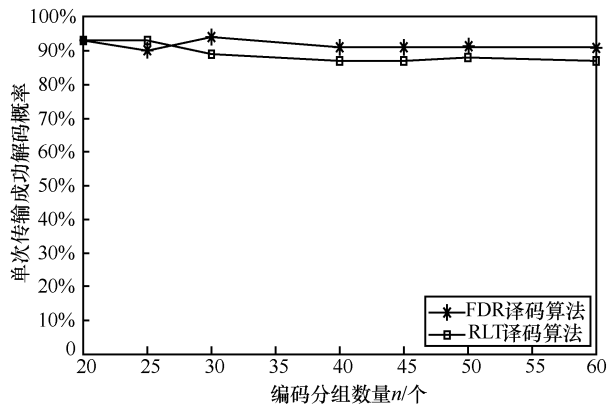


图 9 RLT 译码和 FDR 译码成功概率对比

实验中还统计了采用 FDR 算法在不同原始分组、不同编码分组、不同发送分组频率下的 4×4 组  $N_{Total-trans}$  与  $N_{retrans}$  实验数据，分别如图 10~图 13 所示。图 10 显示了在 0.64 packet/s 发送分组频率下，原始分组数为 25、编码分组数为 32 时统计的 4 组实验数据。图 11 显示了在 0.57 packet/s 发送分组频率下，原始分组数为 30、编码分组数为 38 时统计的 4 组实验数据。图 12 给出在 0.51 packet/s 发送分组频率下，原始分组数为 35、编码分组数为 44 时统计的 4 组实验数据。图 13 给出在 0.70 packet/s 发送分组频率下，原始分组数为 40，编码分组数为 50 时统计的 4 组实验数据。

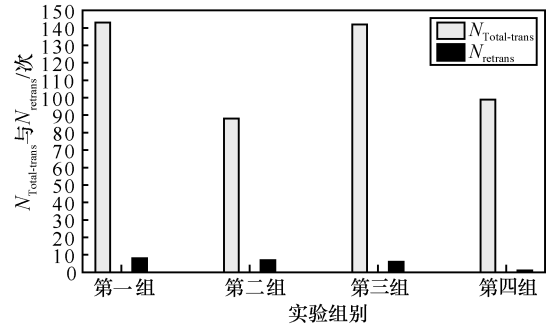


图 10  $k=25$ 、 $n=32$  时的  $N_{Total-trans}$  与  $N_{retrans}$

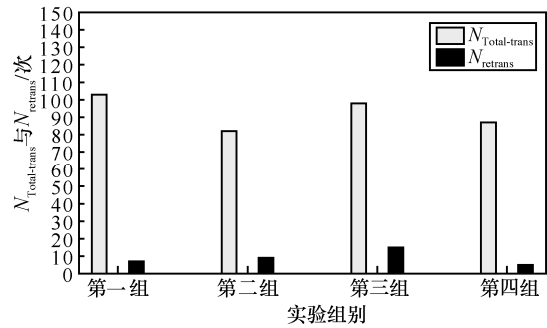


图 11  $k=30$ 、 $n=38$  时的  $N_{Total-trans}$  与  $N_{retrans}$

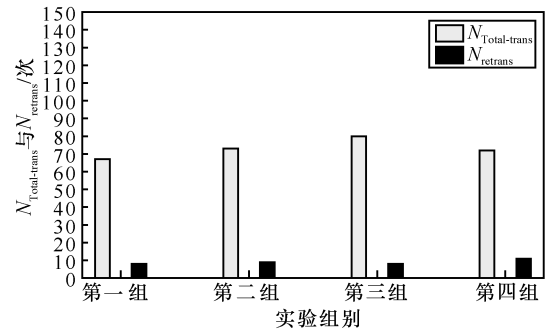


图 12  $k=35$ 、 $n=44$  时的  $N_{Total-trans}$  与  $N_{retrans}$

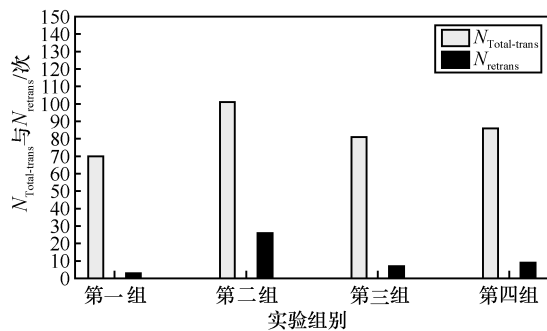


图 13  $k=40$ 、 $n=50$  时的  $N_{Total-trans}$  与  $N_{retrans}$

## 6 结束语

本文提出一种 FDR 算法，消除采用传统译码方式的 RLT 译码算法在收到一定数量编码分组才

开始解码的等待时间, 实现边接收边尝试解码的快速译码方式。此外, 通过编码分组之间的异或运算, 有效增加了度为1的编码分组的产生概率, 不再仅依赖于从发送端获取度为1的编码分组, 在降低传输时延的同时, 通过增加度为1的编码分组出现的概率, 从而提高译码成功率。通过NS3仿真平台在解码成功概率方面与RLT码进行仿真对比, 实验结果显示, FDR算法的解码成功概率普遍高于RLT码, 当编码分组数量为30~40个时, FDR算法的解码成功概率比RLT码高5%, 且FDR算法译码成功率最高可达95%; 当编码分组数量为40~60个时, FDR算法的解码成功概率比RLT码高3%, 且FDR算法的译码成功概率基本保持在89%~90%。实验结果验证了FDR算法对于水声网络可靠传输的有效性和稳健性。

### 参考文献:

- [1] KUN H, ZHIGANG J, HAIFENG S. An efficient and reliable geographic routing protocol based on partial network coding for underwater sensor networks[J]. *Sensors*, 2015, 15(6): 12720-12735.
- [2] DU X, HUANG K, LAN S. LB-AGR: level-based adaptive geo-routing for underwater sensor network[J]. *The Journal of China Universities of Posts and Telecommunications*, 2014, 21(1): 54-59.
- [3] 王燕, 李晴, 张光普, 等. 超短基线水声定位系统相位差模糊算法研究[J]. *电子学报*, 2017(11): 2787-2794.  
WANG Y, LI Q, ZHANG G P, et al. Phase-difference ambiguity resolution for USBL underwater acoustic positioning systems [J]. *Acta Electronica Sinica*, 2017, 45(11): 2787-2794.
- [4] DU X, PENG C, LIU X, et al. Hierarchical code assignment algorithm and state-based CDMA protocol for UWSN[J]. *Wireless Communication over ZigBee for Automotive Inclination Measurement*. *China Communications*, 2015, 12(3): 50-61.
- [5] 蒋鹏, 冯洋, 吴锋. 果蝇启发的水下传感器部署算法[J]. *电子学报*, 2017, 45(6): 1403-1407.  
JIANG P, FENG Y, WU F. Fruit fly inspired underwater sensor network deployment algorithm[J]. *Acta Electronica Sinica*, 2017, 45(6): 1403-1407.
- [6] 张行, 常颖, 宋康. 一种新颖的水声信道参数估计算法[J]. *电子学报*, 2019, 47(2): 509-512.  
ZHANG X, CHANG Y, SONG K. A novel parameter estimation algorithm for underwater acoustic channels[J]. *Acta Electronica Sinica*, 2019, 47(2): 509-512.
- [7] 刘康宁, 赵增华, 王宇. AdaCode: IEEE 802.11n 长距离无线链路自适应编解码算法[J]. *电子学报*, 2018, 46(12): 128-135.  
LIU K N, ZHAO Z H, WANG Y. AdaCode: adaptive codes in IEEE 802.11n wireless long-distance links[J]. *Acta Electronica Sinica*, 2018, 46(12): 128-135.
- [8] 马鸿洋, 范兴奎, 王淑梅, 等. 基于量子隐形传态的水下传感器网络分级加密通信协议[J]. *软件学报*, 2014, 25, 1(1): 39-46.  
MA H Y, FAN X K, WANG S M, et al. Secure hierarchical hybrid encryption communication protocol based on quantum teleportation for underwater sensor networks[J]. *Journal of Software*, 2014, 25, 1(1): 39-46.
- [9] 朱剑, 刘君, 赵海, 等. UWSN 中基于随机游走模型的可靠路由算法研究[J]. *计算机学报*, 2016, 39(5): 1007-1020.  
ZHU J, LIU J, ZHAO H, et al. Research on reliable protocol based on random-walk model in UWSN[J]. *Chinese Journal of Computers*, 2016, 39(5): 1007-1020.
- [10] GUO Z, XIE P, CUI J. On applying network coding to underwater sensor networks[C]//*Proceedings of the 1st ACM International Workshop on Underwater Networks*. Trier: DBLP, 2006: 109.
- [11] LIU B, GARCIN F, REN F. A study of forward error correction schemes for reliable transport in underwater sensor networks[C]//*5th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*. Piscataway: IEEE Press, 2008: 197-205.
- [12] XIE P, ZHOU Z, PENG Z. SDRT: a reliable data transport protocol for underwater sensor networks [J]. *Ad Hoc Networks*, 2010, 8(7): 708-722.
- [13] LIU B, CHEN H, LEI X, et al. Internode distance-based redundancy reliable transport in underwater sensor networks[J]. *EURASIP Journal on Wireless Communications and Networking*, 2010: 1-16.
- [14] MO H, PENG Z, ZHOU Z, et al. Coding based multi-hop coordinated reliable data transfer for underwater acoustic networks: Design, implementation and tests[C]//*Globecom Workshops*. Piscataway: IEEE Press, 2013: 4566-4571.
- [15] 朱文杰, 易本顺, 甘良才, 等. 一种基于模运算和扩展欧几里得定理的喷泉码[J]. *电子学报*, 2017, 45(4): 855-862.  
ZHU W J, YI B S, GAN L C, et al. Novel fountain codes based on modulo and extended Euclidean[J]. *Acta Electronica Sinica*, 2017, 45(4): 855-862.
- [16] 牛芳琳, 李宝明, 陈付亮, 等. 一种改进的基于部分信息喷泉码度分布设计[J]. *电子学报*, 2016, 44(2): 295-300.  
NIU F L, LI B M, CHEN F L, et al. The improved degree distribution for rateless code under partial information[J]. *Chinese Journal of Electronics*, 2016, 44(2): 295-300.
- [17] DU X, LI K, LIU X. RLT code based handshake-free reliable MAC protocol for underwater sensor networks[J]. *Journal of Sensors*, 2016.
- [18] 杜秀娟, 苏毅珊. 水下传感器网络研究[M]. 北京: 科学出版社, 2016.  
DU X J, SU Y S. Research about underwater sensor network[M]. Beijing: Science Press, 2016.
- [19] 焦健, 杨志华, 顾术实. 基于随机置换展开与停止集的LT码联合编译码算法[J]. *通信学报*, 2013, 34(2): 31-39.  
JIAO J, YANG Z H, GU S S. Novel joint encoding/decoding algorithms of LT codes based on random permute edge-growth and stopping set[J]. *Journal on Communications*, 2013, 34(2): 31-39.

### [作者简介]



王丽娟 (1992-), 女, 河北石家庄人, 青海师范大学博士生, 主要研究方向为无线网络与安全。

杜秀娟 (1970-), 女, 河北石家庄人, 博士, 青海师范大学教授、博士生导师, 主要研究方向为无线网络与安全、物联网技术等。

李冲 (1990-), 男, 河北邯郸人, 青海师范大学博士生, 主要研究方向为无线网络与安全。