

# 基于极大后验概率的高容错码速调整恢复算法

刘波, 陆佩忠, 邹艳

(复旦大学计算机科学与工程系, 上海 200433)

**摘要:** 针对高误码信道环境下的数字分接中的码速恢复问题, 提出一种高容错码速恢复算法——高容错预测算法。该算法利用二次群码速调整的先验统计知识以及码速调整指示码的后验反馈信息, 对当前帧的码速调整的预测放在一条连续预测比特序列中考虑, 预测局部最优的序列, 实现码速恢复的高容错性。仿真结果表明, HERP 算法在预测准确性上比国际电信联盟推荐的多数判决法提高近 10 倍。  
**关键词:** 码速调整; 数字分接; 多数判决法; 高容错预测算法

## High Error-Resilient Justification Recovery Algorithm Based on Maximum Posteriori Probability

LIU Bo, LU Pei-zhong, ZOU Yan

(Department of Computer Science and Engineering, Fudan University, Shanghai 200433)

**【Abstract】** To solve the problem of justification recovery in digital demultiplexing stage in the environment of high error channel, a high error-resilient of justification recovery algorithm, High Error-Resilient Prediction(HERP) algorithm is proposed in this paper. HERP algorithm makes use of the justification statistics knowledge of 2-ary digital group and the feedback information from justification indicating code. This algorithm considers the justification prediction of the current frame in a continuous prediction sequence, which can lead to a global optimal prediction sequence and implement high error-resilient in justification. The simulation experiment shows HERP algorithm has improved nearly by ten times in the aspect of error-resilient and accuracy compared with most decision algorithm which is recommended by ITU.

**【Key words】** justification; digital demultiplexing; most decision algorithm; High Error-Resilient Prediction(HERP) algorithm

在正常通信的误码率  $10^{-4}$  水平下, 国际电信联盟(ITU)<sup>[1]</sup> 推荐的多数判决法可以保证正常的码速恢复。当信道误码率在  $10^{-2}$  左右的特殊通信接收环境下, 多数判决法对码速调整预测时经常会误判, 已经不能完成正常的分接<sup>[2]</sup>。针对高误码率的信息截获应用背景, HERP 算法对 Viterbi 算法<sup>[3-4]</sup> 和 MAP 译码算法<sup>[5]</sup> 改进结合后应用于高速 PDH 同步通信中的数字分接领域, 实现码速调整的高容错序列预测。本文的创新点是将纠错译码的思想用于符合国际标准 ITU-G.743 的没有进行纠错编码的数字分接技术。

### 1 多数判决法

图 1 为 1.544 Mb/s 基群复接成二次群后的帧的时隙图, 帧长为 294 bit。根据 ITU-G.743 标准<sup>[1]</sup>,  $C_{j1}, C_{j2}, C_{j3}$  为支路  $j$  的码速调整指示码,  $V_j$  为支路  $j$  的填充比特。  $C_{j1}, C_{j2}, C_{j3}$  为“111”表示码速调整,  $V_j$  插入随机码;  $C_{j1}, C_{j2}, C_{j3}$  为“000”表示无码速调整,  $V_j$  为该支路的信息比特。

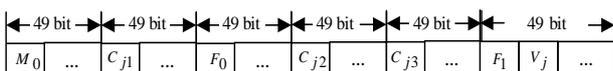


图 1 二次群复接帧时隙图

若码速调整指示码的码长为  $\eta$ , 多数判决法则通过码速调整指示码中是否有大于或等于  $d = (\eta + 1) / 2$  个相同的比特来决定有无码速调整。如果  $\eta$  比特的指示码中出现  $d$  或  $d$  比特以上的误码, 就会导致误判。若信道误码率为  $p$ , 则  $\eta$  比特指示码中出现  $d$  或  $d$  比特以上误码的概率为

$$P_e = C_\eta^d p^d (1-p)^{\eta-d} + C_\eta^{d+1} p^{d+1} (1-p)^{\eta-d-1} + \dots + p^\eta \quad (1)$$

以二次群为例, 基群的传输速率为 1.544 Mb/s, 帧长为 294 bit, 指示码长度  $\eta$  为 3, 根据式(1)可以计算得到以下结果: (1) 当误码率在  $10^{-3}$  时, 平均滑帧时间间隔约为 64 s; (2) 当误码率在  $10^{-2}$  时, 平均不到 1 s 就会出现滑帧现象。因此, 在信息截获背景下, 用常规的多数判决方法无法实现正常数字分接。

## 2 HERP 算法

### 2.1 基本原理

由码速调整的原理和过程<sup>[6]</sup>可知, 当前帧是否需要码速调整与当前帧的前面连续若干帧的码速调整状态具有重要联系。当信道误码率很高时, 误码造成码速调整指示码失效。这时可以通过观察当前帧的前面连续若干帧的码速调整状态来帮助决策当前帧是否需要码速调整。

在二次群的复接帧中, 如果当前帧需要进行码速调整, 预测比特记录为“1”, 否则预测比特记录为“0”, 码速恢复的序列则由预测比特构成一条 01 序列。通过观察标称速率为 1 544 Kb/s 的基群信号复接成二次群的码速调整变化, 发现当前帧的调整与前面  $L$  帧是否调整有重要联系。表 1 为  $L$

**基金项目:** 国家自然科学基金资助重大项目(90204013); 国家自然科学基金资助项目(60673082)

**作者简介:** 刘波(1983-), 男, 硕士研究生, 主研方向: 纠错编码, 信息安全; 陆佩忠, 教授、博士生导师; 邹艳, 博士

**收稿日期:** 2007-12-20 **E-mail:** boliu.fudan@gmail.com

取 4 的实验结果, 总共实验的帧数为 59 608 帧。表 1 中的第 1 列记录是在当前帧的前 4 帧的码速调整状态, 第 2 列和第 3 列分别记录实验中观察到当前帧不需码速调整的帧数和需要调整的帧数。

表 1 二次群码速调整观察的实验结果

前 4 帧的状态	当前帧不调整	当前帧调整
0000	0	124
0001	4 148	167
0010	10 573	4 556
0011	167	0
0100	4 148	10 982
0101	4 556	0
0110	167	0
1000	124	4 191
1001	10 981	0
1010	4 557	0
1100	167	0

把表 1 的统计结果转化为状态转移图可以得到图 2。其中, 图 2 的 11 个状态对应表 1 的 11 项前 4 帧码速调整状态, 每个状态箭头上的权值对应表 1 中该状态不调整和调整的帧数。

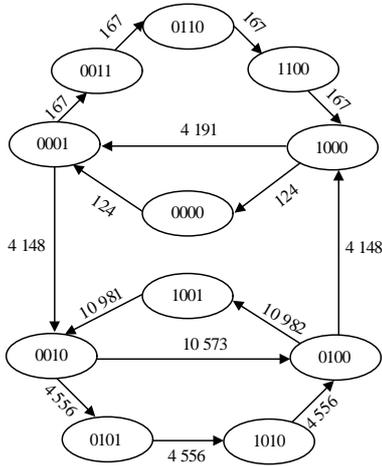


图 2 状态转移图

从表 1 的统计结果以及图 2 的状态图可知: (1) 码速调整的状态小于  $2^4$  项; (2) 在前 4 帧的码速调整状态确定时, 其中有 7 个状态可以完全确定当前帧是否需要调整。在其他状态中虽然不能由前面的状态确定当前帧是否需要调整, 但当前帧调整和不调整的帧数相差较大。因此, 在信道误码率较高时, 完全可以把图 2 中的先验统计信息用到码速恢复时对码速调整的预测, 来增强对码速调整预测的容错性。

## 2.2 算法的提出

首先以图 2 为例定义以下符号:

$s_i$  表示从当前帧开始计数后面第  $i$  帧对应的前 4 帧状态,  $i \in \{1, 2, \dots, n\}$ ,  $s_i \in S$ ,  $S$  为所有前 4 帧状态的集合。例如  $s_1$  表示当前帧的前 4 帧状态, 若为“0001”。 $s_2$  表示下一帧对应的前 4 帧状态, 如果当前帧的预测比特为“0”, 则  $s_2$  为“0010”, 如果当前帧的预测比特为“1”, 则  $s_2$  为“0011”。下文中把“从当前帧开始计数的后面第  $i$  帧”简称为“第  $i$  帧”, “前 4 帧状态”简称为“状态”。

$P_{ij}^{s_i}$  表示第  $i$  帧状态为  $s_i$  时预测比特为  $j$  的先验统计概率,  $i \in \{1, 2, \dots, n\}$ ,  $j \in \{0, 1\}$ 。例如  $s_i$  为“0001”,  $j$  取 0, 则

根据图 2 可知,  $P_{i0}^{0001} = 4148/(4148+167)$ 。

$p(Q_{ij} | z = z_i)$  表示当第  $i$  帧的码速调整指示码中有  $z_i$  个 0 时, 第  $i$  帧的预测比特为  $j$  的置信概率,  $i \in \{1, 2, \dots, n\}$ ,  $j \in \{0, 1\}$ 。具体定义为

$$\begin{cases} p(Q_{i0} | z = z_i) = p^{\eta-z_i}(1-p)^{z_i} \\ p(Q_{i1} | z = z_i) = p^{z_i}(1-p)^{\eta-z_i} \end{cases} \quad (2)$$

其中,  $z_i$  表示第  $i$  帧的码速调整指示码中 0 的个数,  $\eta$  为码速调整指示码的长度, 根据 ITU-G.743 建议<sup>[1]</sup>, 码速调整指示码的长度  $\eta$  为 3, 则  $z_i \in \{0, 1, 2, 3\}$ ,  $i \in \{1, 2, \dots, n\}$ 。 $p$  为信道误码率。

$W_{ij}^{s_i}$  表示第  $i$  帧状态为  $s_i$  的预测比特为  $j$  的概率,  $j \in \{0, 1\}$ , 定义为

$$W_{ij}^{s_i} = P_{ij}^{s_i} \times p(Q_{ij} | z = z_i) \quad (3)$$

$U_{d^n}^{s_1}$  表示起始状态为  $s_1$ , 经过  $n$  帧预测得到预测序列  $j_1, j_2, \dots, j_n$  的概率, 定义为

$$U_{d^n}^{s_1} = \prod_{i=1}^n W_{ij_i}^{s_i} \quad (4)$$

由于  $j_i \in \{0, 1\}$ , 因此预测序列  $j_1, j_2, \dots, j_n$  最多有  $2^n$  条。其中,  $d^n$  表示起始状态为  $s_1$ , 经过  $n$  帧预测得到预测序列的个数, 所以  $d^n \leq 2^n$ 。

若图 2 中状态的下一个状态多于一个, 称该状态为分支状态; 若图 2 中状态的下一个状态唯一确定, 称该状态为必然状态。例如图 2 中状态“0001”是分支状态, 状态“0000”是必然状态。由此可得图 2 中含有分支状态的集合为  $\Phi = \{0001, 0010, 0100, 1000\}$ , 除去集合  $\Phi$  中的状态的其他状态均为必然状态。

下面以图 2 为例子, 描述 HERP 算法的步骤:

(1) 根据图 2 先验统计信息, 创建预测表  $T$ ,  $T$  中存储起始状态为所有分支状态  $s_1$ , 路径长度  $n$  的所有可能预测序列的概率值  $U_{d^n}^{s_1}$ 。在计算概率值  $U_{d^n}^{s_1}$  时, 由于不能确定在第  $i$  帧的码速调整指示码中 0 的实际反馈个数  $z_i$ , 因此在预测表  $T$  中存储  $z_i$  取  $\{0, 1, 2, 3\}$  所有情况下的概率值。信道误码率  $p$  可以根据信噪比或者同步码中的误码情况估算。

(2) 取当前帧的状态  $s_0$ , 如果  $s_0 \in \Phi$ , 则转到(3); 否则根据图 2 中必然状态  $s_0$  对应的下一个状态  $s_0'$  产生当前帧的预测比特“0”或“1”, 同时更新当前帧的前 4 帧状态  $s_0 = s_0'$ 。转到(2)。

(3) 把状态  $s_0$  作为起始状态, 以当前帧以及之后连续共  $n$  帧对应码速调整指示码中 0 的个数  $z_1, z_2, \dots, z_n$  作为参数查询预测表  $T$ , 得到起始状态  $s_0$  路径长度为  $n$  在  $z_i$  取  $z_1, z_2, \dots, z_n$  时所有预测序列的概率值  $\Delta = \{U_{d^n}^{s_0}\}$ 。

(4) 记集合  $\Delta$  中概率值最大的为  $U_{\max}^{s_0}$ 。如果  $U_{\max}^{s_0}$  第 1 个预测比特为“0”, 则当前帧预测比特为“0”; 如果  $U_{\max}^{s_0}$  第 1 个预测比特为“1”, 则当前帧的预测比特为“1”。把  $s_0$  中的后 3 位加上当前帧的预测比特组成下一帧的状态  $s_0'$ , 并更新当前帧的状态  $s_0 = s_0'$ 。转到(2), 直到处理所有比特。

## 2.3 算法分析

与多数判决法相比, HERP 算法主要在以下 2 个方面进行了改进: (1) 对当前帧的码速调整的预测不仅利用当前帧的码速调整指示码的信息, 而且结合先验统计知识来加强预测

的准确性和容错性；(2)对当前帧的码速调整的预测不是只考虑当前帧的预测比特的概率分布，而是把当前帧的预测比特放到一条连续的预测比特序列中来考虑，寻求一条局部最优的预测序列。

下面对 HERP 算法的算法复杂性进行分析，记一次乘法运算和一次比较运算的代价均为 1。由算法步骤(1)以及式(2)~式(4)可知构造预测表  $T$  最多需要代价为

$$C_{Tn} = \lambda \times 2^n \times 4^n \times (2 \times n - 1) \quad (5)$$

其中， $n$  取小于 6 的正整数。

算法对当前帧预测时，如果当前帧的状态  $s_0$  为必然状态时，则根据图 2 的状态转移图得到当前帧的预测比特，只需一次比较运算；否则，需要以  $s_0$  和  $z_1, z_2, \dots, z_n$  为参数查询预测表，由于预测表  $T$  已经存储所有可能情况的概率值，因此这里不需要其他的计算。然后从起始状态为  $s_0$  的所有预测序列在  $z_i$  取  $z_1, z_2, \dots, z_n$  的概率值中选择概率值最大的预测序列，因为每个分支状态对应的预测序列数目不超过  $2^n$ ，所以至多需要  $2^n - 1$  次比较运算。若  $\alpha$  为分支状态、 $\beta$  为必然状态在预测过程中分别出现的次数，则对每帧进行预测的平均代价为

$$C_{Fn} = \frac{\alpha \times (2^n - 1) + \beta}{\alpha + \beta} \quad (6)$$

由于必然状态的存在，在实际应用中预测序列的数目要远小于  $2^n$ 。以图 2 为例，当  $n=5$  时，经计算可知分支状态对应的预测序列的数目最多为 11 条。把图 2 中出现的必然状态和分支状态的次数代入式(6)，可以得到图 2 预测一帧需要的代价  $C_{F5} = 8.8$ ，相当于不到 9 次比较运算的代价。由于预测表的构造只需计算一次，分摊到每一帧上代价非常小，因此 HERP 算法在对每一帧码速调整预测时的平均代价大约相当于 9 次比较运算的代价。

由以上分析可以知道，HERP 算法通过创建预测表的方式极大地降低了预测时的计算复杂度。对每一帧的预测时不需要做乘法运算，平均的代价相当于 9 次比较运算。由于每个复接帧有数千个比特，因此平均预测的代价分摊到每个输入比特的时间内所需要的计算量非常小。因此该算法也易于用软硬件实现。

### 3 实验仿真与结果

实验仿真有 2 个目的：(1)确定 HERP 算法中的预测序列长度  $n$  为何值时，该算法预测的效果稳定；(2)比较多数判决法和 HERP 算法对码速调整预测的错误率。实验以 ITU-G.743 建议<sup>[1]</sup>为标准构造仿真环境。

实验仿真的步骤如下：

(1)由计算机随机生成长度为 50 万帧的码速调整源序列  $q_s$ ，使得该序列在分布上满足图 2 的状态转移图的概率分布。

(2)按照“0”扩张成“000”和“1”扩张成“111”的方式，扩张源序列  $q_s$  得到  $q_e$ ，其中，“000”和“111”表示帧的码速调整指示码。

(3)以误码率  $p$  往扩张后的序列  $q_e$  叠加误码得到含有误码的序列  $q_r$ ，考虑到信息截获的实际应用背景，实验中的信道误码率  $p$  的取值范围为  $[1 \times 10^{-3}, 2 \times 10^{-2}]$ ，间距为  $10^{-3}$ 。

(4)在  $n$  取不同值的情况下，对序列  $q_r$  应用 HERP 算法进行码速恢复，得到相应的恢复序列  $q_m$ 。考虑到式(5)计算的复杂性与  $n$  的指数关系和实际的预测效果，实验中的  $n$  的取值

为 {3,4,5}。

(5)对序列  $q_r$  应用多数判决法进行码速恢复，得到恢复序列  $q_d$ 。

(6)比较码速调整源序列  $q_s$  和恢复序列  $q_m$  和  $q_d$ ，计算预测错误率  $e_m$  和  $e_d$ 。

(7)重复步骤(1)~步骤(6)3 次，分别得到预测错误率  $e_m$  和  $e_d$ ，取其平均值得到  $\bar{e}_m$  和  $\bar{e}_d$ 。

照上述实验步骤可以得到 HERP 算法预测错误率与  $n$  的关系，如图 3 所示。由图 3 的曲线图可知：(1)HERP 算法的预测错误率随  $n$  的增加而减小，两者基本满足反比关系；(2) $n$  取 5 的效果比  $n$  取 4 略好一些，但两者的预测错误率相差很小，因此可以认为在  $n$  取 4 或 5 时，算法预测效果基本稳定。

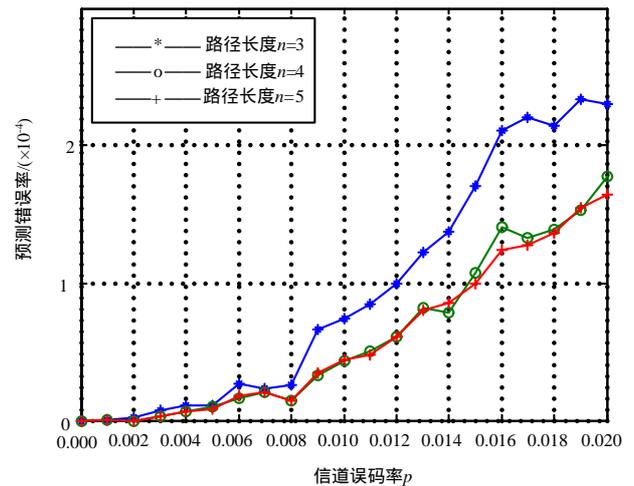


图 3  $n$  取不同值的 HERP 算法预测错误率曲线图

在实验仿真中计算了多数判决法预测错误率的实验值，其实多数判决法预测错误率的理论值可通过式(7)计算得到

$$E_n = C_n^{(\eta+1)/2} p^{(\eta+1)/2} (1-p)^{(\eta-1)/2} + \dots + C_n^\eta p^\eta \quad (7)$$

其中， $\eta$  为码速调整指示码的长度，在本实验中  $\eta = 3$ 。

图 4 为 HERP 算法 ( $n$  取 5) 与多数判决法 (实验值、理论值) 的预测错误率比较的实验结果，从图中可以知道 HERP 算法的预测错误率整体上要比多数判决法低很多。

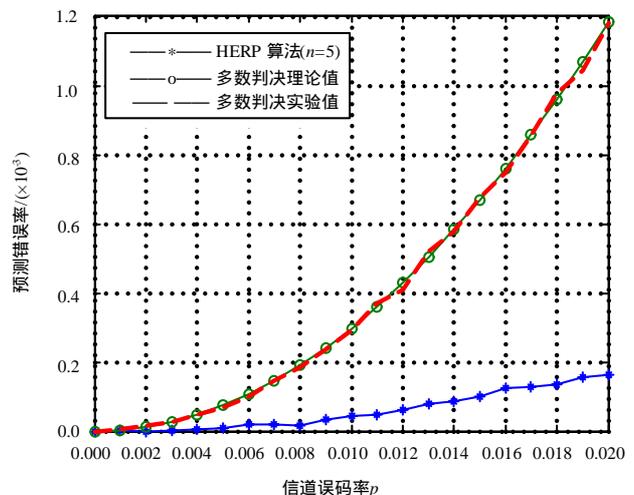


图 4 HERP 算法与多数判决法预测错误率曲线图

若定义多数判决法 (实验值) 和 HERP 算法两者预测错误率的比率为

(下转第 123 页)