

## 一种基于子集约束的协议首部纠错算法

王晓梅<sup>①</sup> 范亮<sup>\*①</sup> 陈彦<sup>①</sup> 洪先强<sup>①</sup>

(解放军信息工程大学信息工程学院 郑州 450002)

**摘要:**针对无线网络数据的协议首部容易出错问题,该文在研究基于循环冗余校验的协议首部纠错算法的基础上,提出一种基于子集约束的纠错算法。该算法利用接收比特的置信度信息以接收向量为中心构建约束子集,从而缩小运算搜索范围,克服此前算法运算复杂度高的缺陷。随后,结合无线信号类型与信道模型,对算法的测试长度参数的取值范围进行了理论分析和实验验证。仿真结果表明,对于不同信噪比的无线信号,该算法可通过改变测试长度来调节约束子集大小,在保证较好性能条件下有效地降低运算开销,具有较强的实际应用价值。

**关键词:**无线通信;无线多媒体;协议首部纠错;校验字段

**中图分类号:** TP393.0

**文献标识码:** A

**文章编号:** 1009-5896(2015)08-2014-07

**DOI:** 10.11999/JEIT141574

## Header Recovery Algorithm Based on Subset Constraint

Wang Xiao-mei<sup>①</sup> Fan Liang<sup>①</sup> Chen Yan<sup>①</sup> Hong Xian-qiang<sup>①</sup>

(Institute of Information Engineering, PLA Information Engineering University, Zhengzhou 450002, China)

**Abstract:** For the protocol headers of wireless network data prone to errors, this paper puts forward with a bit-flip subset restriction header recovery algorithm after studying the one based on Cyclic Redundancy Check (CRC). A constraint subset of the received vector centric is set up to narrow the search space by exploiting the confidence information of each bit, overcoming the defect of high complexity of the former header recovery algorithm. Then, the theoretical analysis and experimental verification about the value range of the test vector's length are done combining the models of wireless signal and wireless channel. The simulation results show that this method can maintain the well performance with a low computing cost, adjusting the test vector's length towards wireless signals with different Signal to Noise Ratio (SNR).

**Key words:** Wireless communication; Wireless multimedia; Protocol header recovery; Verifying fields

### 1 引言

随着人们对方便、快捷、相对自由的无线通信越来越热衷,无线技术也因此取得飞速发展。但无线通信始终处于开放环境容易受到各类干扰,传输误比特率相对较高<sup>[1]</sup>,因而一直制约了它的应用。针对这一问题,文献[2,3]提出在反馈重传机制下通过接收端对接收数据进行处理实现对含错的分组数据恢复,减少重传次数来提高接收效率。然而,对于一些实时要求高或只能单向通信的特殊场合反馈重传机制无法应用。于是一些学者们另辟蹊径对网络协议的冗余展开研究,提出了针对协议首部纠错的容错接收算法<sup>[4-9]</sup>,利用冗余提高系统的容错能力。

针对无线通信协议中控制字段的取值存在着冗余特点,文献[4]提出了在高斯信道背景下基于似然

概率的协议首部纠错算法(Robust Header Recovery, RHR),在一定程度上实现了对无线协议首部的纠错。无独有偶,文献[5,6]出于同样的想法,并考虑到无线终端的运行成本,给出了在二进制对称信道条件下基于最小距离的协议首部纠错方法(Min Distance header recovery, Min DIS)。该方法用比特向量的二元域运算代替了RHR算法中的概率运算,具有较强的可实现性,但也造成了纠错性能的损失。但上述两种算法只是研究了字段自身的冗余,并没有涉及字段间所存在的冗余。文献[7~9]对协议中校验字段携带的冗余信息进行了探索,提出了联合循环冗余校验的协议首部纠错算法(Robust Cyclic Redundancy Check header recovery, Robust CRC),通过字段间的冗余信息提高纠错性能。该算法的缺点是运算复杂度随校验长度成指数式的增长,纠错性能也会随着校验长度的增加而下降。而实际通信协议中校验字段的校验范围往往较长,该算法的实现和应用是一个难题。尽管借助维特比格

2014-12-10 收到, 2015-04-07 改回, 2015-06-08 网络优先出版

国家安全重大基础研究(6131482013)资助课题

\*通信作者: 范亮 fanly6@163.com

型译码形式的计算方法可降低运算复杂度，但复杂度依旧较高，同时也引入存储空间开销大的问题。

而且文献[1,4-9]中提到为了最大限度地保留接收信号的有用信息，所有处理过程都应该以软信息的形式输入输出，如此则会进一步增加纠错算法的运算成本，限制了其实用性。为此，本文借鉴 Chase 译码算法<sup>[10,11]</sup>的思想，在 Robust CRC 纠错算法基础上提出了一种约束子集的协议首部纠错算法(Bit-Flip subset constraint CRC, BF-CRC)。该算法借助每比特的置信度信息对接收信号进行预先处理，建立一个以接收向量为中心的约束子集，通过调整约束子集的大小来维持较好的纠错性能，最终实现算法的搜索范围地缩小，有效地降低算法的运算复杂度，解决了 Robust CRC 算法无法实用的问题。

## 2 子集约束的协议首部纠错算法

### 2.1 基于校验关系协议首部纠错算法概述

网络协议字段依据其在通信过程中不同特性可分为<sup>[8]</sup>：(1)固定字段  $\mathbf{K}$ ：实现通信控制、帧同步等功能的取值固定的字段。(2)可推测字段  $\mathbf{P}$ ：根据前后时刻传输的分片数据或上下层关系可推测出取值的字段。因为字段  $\mathbf{K}$  和字段  $\mathbf{P}$  在纠错时都认为是已知的，所以后续叙述中统一描述为字段  $\mathbf{K}$ 。(3)未知字段  $\mathbf{U}$ ：在数据传输过程中对与通信链接起到重要作用的字段，其取值空间的大小可以根据协议内的约束关系进行有效的压缩。鉴于其对通信的重要作用，协议首部纠错实质就是对该部分进行恢复。(4)不关心字段  $\mathbf{O}$ ：该字段的取值无法在当前协议层中解析，或者其取值对通信的影响不大；(5)校验字段  $\mathbf{C}$ ：在通信协议中起到差错控制作用的字段，该部分完全属于通信时额外引入的冗余信息。当发送数据为  $(\mathbf{k}, \mathbf{u}, \mathbf{o}, \mathbf{c})$ ，对应的接收数据为  $(\mathbf{y}_k, \mathbf{y}_u, \mathbf{y}_o, \mathbf{y}_c)$  时，根据最大后验概率准则可知，未知字段  $\mathbf{U}$  的最佳估计为

$$\hat{\mathbf{u}} = \arg \max_{\mathbf{u} \in \Omega_U} P(\mathbf{u} | \mathbf{y}_k, \mathbf{y}_u, \mathbf{y}_o, \mathbf{y}_c) \quad (1)$$

由于字段  $\mathbf{K}$  已知，即  $P(\mathbf{u} | \mathbf{y}_k, \mathbf{y}_u, \mathbf{y}_o, \mathbf{y}_c) \propto P(\mathbf{u} | \mathbf{y}_u, \mathbf{y}_o, \mathbf{y}_c, \mathbf{k})$ 。假设信道满足无记忆的高斯信道时，则有

$$\begin{aligned} P(\mathbf{u} | \mathbf{y}_u, \mathbf{y}_o, \mathbf{y}_c, \mathbf{k}) &\propto P(\mathbf{u}, \mathbf{y}_u, \mathbf{y}_o, \mathbf{y}_c, \mathbf{k}) \\ &= P(\mathbf{u})P(\mathbf{y}_u | \mathbf{u}) \cdot P(\mathbf{y}_o, \mathbf{y}_c | \mathbf{k}, \mathbf{u}) \\ &= P(\mathbf{u})P(\mathbf{y}_u | \mathbf{u}) \cdot \sum_{\mathbf{o}} P(\mathbf{o})P(\mathbf{y}_o, \mathbf{y}_c | \mathbf{k}, \mathbf{u}, \mathbf{o}) \\ &= P(\mathbf{u})P(\mathbf{y}_u | \mathbf{u}) \cdot \Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u}) \end{aligned} \quad (2)$$

$$\Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u}) = \sum_{\mathbf{o}} P(\mathbf{o})P(\mathbf{y}_o, \mathbf{y}_c | \mathbf{k}, \mathbf{u}, \mathbf{o}) \quad (3)$$

将上述式(2)代入式(1)，则估计算子可以转化为

$$\hat{\mathbf{u}} = \arg \max_{\mathbf{u} \in \Omega_U} P(\mathbf{u})P(\mathbf{y}_u | \mathbf{u}) \cdot \Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u}) \quad (4)$$

设协议首部中各个字段的比特长度分别为  $l_k$ ,  $l_u$ ,  $l_o$  和  $l_c$ 。由式(3)可知，计算  $\Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u})$  需要遍历字段  $\mathbf{O}$  的所有可能取值并求和，该项操作的运算复杂度为  $O(2^{l_o})$ 。当协议字段中不关心字段相对较长，式(4)中基于最大后验准则的最佳估计则会因为巨大运算开销而难以实现，因此亟需能够有效降低运算复杂度的纠错算法。

为此，文献[8]借鉴线性分组码的格型译码<sup>[12]</sup>思想展开了相应的研究。设 CRC 校验函数为  $f(\cdot)$ ，字段  $\mathbf{K}$  和字段  $\mathbf{U}$  分别取值分别为  $\mathbf{k}$  和  $\mathbf{u}$ ，其他字段置为 0 所构成的向量经过  $f(\cdot)$  运算后的校验值记为  $f(\mathbf{k}, \mathbf{u}, \_)$ 。依据校验函数分解的特性有式(5)成立。

$$\mathbf{c} = f(\mathbf{k}, \mathbf{u}, \mathbf{o}) = f(\mathbf{k}, \mathbf{u}, \_) \oplus f(\mathbf{o}) \quad (5)$$

设  $s_i$  为校验状态取值， $\mathbf{y}_o^j$  为接收向量  $\mathbf{y}_o$  的第  $j$  比特， $\pi_j$  为  $\mathbf{O}$  字段的第  $j$  比特  $\mathbf{o}^j$  对应的校验值， $V_{s_i}(j)$  为在第  $j$  比特的校验状态满足  $\mathbf{S}(j) = s_i$  的条件下，接收到后续比特  $(\mathbf{y}_o^{j \leq k \leq l_o}, \mathbf{y}_c)$  的条件概率。经过推导有式(6)~式(8)成立。

$$\begin{aligned} \Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u}) &= P(\mathbf{y}_o, \mathbf{y}_c | \mathbf{S}(0)) \\ &= f(\mathbf{k}, \mathbf{u}, \_) = V_{f(\mathbf{k}, \mathbf{u}, \_)}(0) \end{aligned} \quad (6)$$

$$V_{s_i}(j) = P(\mathbf{y}_o^{j \leq k \leq l_o}, \mathbf{y}_c | \mathbf{S}(j) = s_i) \quad (7)$$

$$\begin{aligned} V_{s_i}(j-1) &= P(\mathbf{o}^{j-1} = 0)V_{s_i}(j)P(\mathbf{y}_o^{j-1} | 0) \\ &\quad + P(\mathbf{o}^{j-1} = 1)V_{s_i \oplus \pi_j}(j)P(\mathbf{y}_o^{j-1} | 1) \end{aligned} \quad (8)$$

式(7)可借用格型图以逐比特迭代的方式实现计算<sup>[8]</sup>，于是计算  $\Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u})$  复杂度降为  $O(l_o 2^{l_c})$ 。此方法为基于校验的纠错算法的实现提供一个解决思路，但却存在着  $O(2^{l_c})$  数量级的存储开销的问题。如果校验字段长度较长，上述算法的计算复杂度和空间复杂度也会非常高，例如对于实际的校验字段为 4 Byte 的情形则需要  $2^{32}$  个 double 类型的存储单元(大约为 16 GByte)来存储校验状态概率，这样的实现压力对于当前无线终端设备仍旧是无法承担的。

### 2.2 子集约束的协议首部纠错算法

根据高斯信道具无记忆性，且式(3)等号右侧的后半部分的条件概率  $P(\mathbf{y}_o, \mathbf{y}_c | \mathbf{k}, \mathbf{u}, \mathbf{o})$  只与错误图样  $\varepsilon = (\mathbf{y}_o \oplus \mathbf{o}, \mathbf{y}_c \oplus f(\mathbf{k}, \mathbf{u}, \mathbf{o})) = (\varepsilon_o, \varepsilon_c)$  有关。因此将式(3)可变换为

$$\begin{aligned}
\Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u}) &= \sum_{\mathbf{o}} P(\mathbf{o}) P(\mathbf{y}_o, \mathbf{y}_c | \mathbf{k}, \mathbf{u}, \mathbf{o}) \\
&= \sum_{\mathbf{o}} P(\mathbf{o}) P(\mathbf{y}_o, \mathbf{y}_c | \mathbf{o}, \mathbf{c}) \\
&= \sum_{\mathbf{o}} P(\mathbf{o}) P(\mathbf{o} \oplus \varepsilon_o | \mathbf{o}) P(f(\mathbf{k}, \mathbf{u}, \mathbf{o}) \\
&\quad \oplus \varepsilon_c | f(\mathbf{k}, \mathbf{u}, \mathbf{o})) \quad (9)
\end{aligned}$$

联合式(5)有  $\mathbf{y}_c = \mathbf{c} \oplus \varepsilon_c = f(\mathbf{k}, \mathbf{u}, \_) \oplus f(\mathbf{o}) \oplus \varepsilon_c$  成立, 进一步则得  $\mathbf{y}_c \oplus f(\mathbf{k}, \mathbf{u}, \_) = f(\mathbf{o}) \oplus \varepsilon_c$ 。为了方便, 记  $\tilde{\mathbf{c}} = f(\mathbf{o})$  和  $\tilde{\mathbf{y}}_c = f(\mathbf{k}, \mathbf{u}, \_) \oplus \mathbf{y}_c = \tilde{\mathbf{c}} \oplus \varepsilon_c$ , 将它们代入式(9)可得到

$$\begin{aligned}
\Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u}) &= \sum_{\mathbf{o}} P(\mathbf{o}) P(\mathbf{o} \oplus \varepsilon_o | \mathbf{o}) P(\tilde{\mathbf{c}} \oplus \varepsilon_c | \tilde{\mathbf{c}}) \\
&= \sum_{\mathbf{o}} P(\mathbf{o}) P(\mathbf{y}_o, \tilde{\mathbf{y}}_c | \mathbf{o}, \tilde{\mathbf{c}}) \quad (10)
\end{aligned}$$

上述概率本质上可以理解为接收码字  $(\mathbf{y}_o, \tilde{\mathbf{y}}_c)$  的出现概率, 即为字段  $\mathbf{O}$  所有可能取值  $\mathbf{o}$  经过 CRC 校验编码后得到发送码字  $(\mathbf{o}, \tilde{\mathbf{c}})$ , 再经过信道在接收端变为特定接收码字  $(\mathbf{y}_o, \tilde{\mathbf{y}}_c)$  的概率。由式(10)可知, 实现  $\Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u})$  的计算需要遍历字段  $\mathbf{O}$  取值空间中的每一个元素, 则正如前面提到的因运算开销太大而难以实现。实际上分析可知在接收到某具体向量已知时, 字段  $\mathbf{O}$  取值空间中各个取值的转移概率  $P(\mathbf{o} \oplus \varepsilon_o | \mathbf{o})$  并不相等, 因此如果能够剔除那些概率接近 0 的取值则可以有效缩小遍历空间的大小, 降低运算复杂度。通过学习发现在信道编码中已有学者采用类似思路提出了利用接收比特置信度信息来缩小搜索空间的 Chase 译码<sup>[10,11]</sup>算法, 本文将结合该算法给出一种易于实现的基于子集约束纠错方法。

以接收向量为中心, 以  $r$  为半径则可在字段  $\mathbf{O}$  的取值空间  $\Omega_o$  中形成一个的子集  $\Omega'_o$ , 称为约束子集。子集  $\Omega'_o$  与原  $\mathbf{o}$  的取值空间的关系可分为如图 1 中所示两种情形<sup>[13]</sup>。情形(1): 当噪声干扰较小, 发送向量仍处于约束子集内时, 式(10)中原需对整个空间的求和则可以通过对约束子集的求和予以近似, 如图 1(a)所示。而且信噪比越高则处于约束子集外被忽略的概率越小, 上述近似就越准确。情形(2): 当噪声干扰较大, 发送向量处于约束子集外时, 约束子集的求和概率则与原空间的求和概率误差较大, 如图 1(b)所示。当信噪比越低, 则集合外取值所引起近似误差就越大, 甚至会出现无法正确近似的情况。但此时可以通过增加子集的半径使其最终转换成情形(1), 借此来保证上述方法的近似性能。在正常的通信情形下一般都有错误概率远小于 1 的条件成立, 此时一般只需利用子集约束的构建方法设计一个半径较小的子集即可较好地实现对  $\Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u})$  的近似计算。

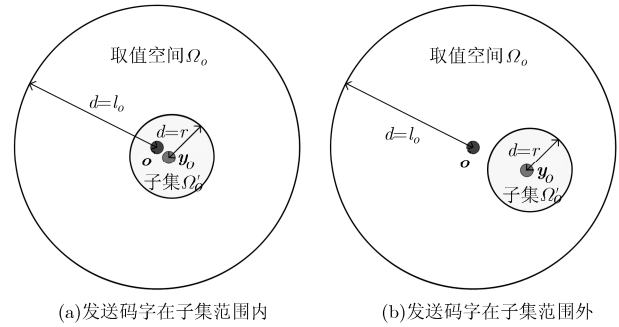


图 1 对于发送为  $\mathbf{o}$  接收数据为  $\mathbf{y}_o$  时, 受到不同程度噪声干扰所对应情形

将  $\mathbf{O}$  字段中置信度较小的比特标识出来, 它们翻转与否分别用 0 和 1 予以表示, 其中 0 表示为不翻转; 1 表示翻转。由上构成的二进制向量称为测试向量, 用  $\mathbf{e}$  表示。  $(\mathbf{y}_o + \mathbf{e})$  表示为原始接收向量  $\mathbf{y}_o$  中对应位置进行相应翻转后的值。根据  $\mathbf{e}$  不同的取值,  $(\mathbf{y}_o + \mathbf{e})$  可以在  $\mathbf{y}_o$  的附近形成一个约束子集。设因置信度较小而翻转的比特的最大个数为  $\delta$ , 称为测试长度。根据定义可知测试长度  $\delta$  描述了约束子集内的元素与硬判决后的接收向量  $\mathbf{y}_o$  的最大汉明, 当以汉明距离表示  $\Omega_o$  中的元素与  $\mathbf{y}_o$  的距离时, 约束子集半径  $r$  与测试长度  $\delta$  是一致的。

根据子集约束的方法可将式(9)近似为

$$\begin{aligned}
\Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u}) &= \sum_{\mathbf{o}} P(\mathbf{o}) P(\mathbf{y}_o, \tilde{\mathbf{y}}_c | \mathbf{o}, \tilde{\mathbf{c}}) \\
&\approx \sum_{(\mathbf{y}_o + \mathbf{e}, f(\mathbf{y}_o + \mathbf{e}))} P(\mathbf{y}_o, \tilde{\mathbf{y}}_c | \mathbf{o}, \tilde{\mathbf{c}}) P(\mathbf{o}) \\
&= \sum_{(\mathbf{y}_o + \mathbf{e}, f(\mathbf{y}_o + \mathbf{e}))} P(\mathbf{y}_o | \mathbf{o}) P(\tilde{\mathbf{y}}_c | \tilde{\mathbf{c}}) P(\mathbf{o}) \quad (11)
\end{aligned}$$

因测试长度通常取值较小, 所对应的需遍历的范围也会较小, 于是算法的运算复杂度下降为  $O(2^\delta)$ , 小于 Robust CRC 算法的  $O(l_o 2^k)$ , 而且该算法不需要同时存储所有校验状态取值的概率所以存储的需求也较低, 保证了算法的可实现性。最后, 将式(10)代入式(4)中可得到对于未知字段  $\mathbf{U}$  的估计。

$$\left. \begin{aligned}
\hat{\mathbf{u}} &= \arg \max_{\mathbf{u} \in \Omega_U} P(\mathbf{u}) P(\mathbf{y}_u | \mathbf{u}) \cdot \Phi'(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u}) \\
\Phi'(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u}) &= \sum_{\Omega'_o} P(\mathbf{y}_o | \mathbf{o}) P(\tilde{\mathbf{y}}_c | \tilde{\mathbf{c}}) P(\mathbf{o}) \\
\Omega'_o &= \{\mathbf{y}_o + \mathbf{e}_0, \mathbf{y}_o + \mathbf{e}_1, \mathbf{y}_o + \mathbf{e}_2, \dots\}
\end{aligned} \right\} \quad (12)$$

设一个接收向量的联合概率  $P(\mathbf{y}_o, \mathbf{y}_c | \mathbf{k}, \mathbf{u}, \mathbf{o})$  的计算消耗为  $M$ , 一次 CRC 校验的计算消耗为  $C$ 。考虑在 BF-CRC 算法中需要快速计算候选数据  $(\mathbf{y}_o + \mathbf{e})$  的校验值, 可以按格雷码<sup>[14]</sup>的次序对测试向量  $\mathbf{e}$  进行遍历。这样在计算新的候选数据的校验值时, 只需要将上一次结果和变化的比特对应的校验

值进行异或运算即可。于是在协议首部中各个字段的比特长度分别为  $l_k, l_u, l_o$  和  $l_c$ ，以及测试长度为  $\delta$  的条件下，最大后验纠错算法 (Maximum A Posteriori, MAP), Robust CRC 以及 BF-CRC 各自在计算  $\Phi(\mathbf{y}_o, \mathbf{y}_c, \mathbf{k}, \mathbf{u})$  所需理论上的运算开销如表 1 所示。

表 1 各算法理论上运算消耗

	联合概率计算消耗 $M$	CRC 校验计算消耗 $C$	累计运算消耗
MAP	$2^b$	$2^b$	$2^b(M + C)$
Robust CRC	$l_o 2^b$	$l_o$	$l_o 2^b M + l_o C$
BF-CRC	$2^\delta$	$\delta + 1$	$2^\delta M + (\delta + 1)C$

接下来以不关心字段为 5 bit，测试向量长度为 2 bit 的情形为例子进行说明。设在接收向量的各个比特对应的置信度 (即为接收信号的测量信息<sup>[10]</sup>) 分别为  $\{1.2, 0.2, -0.1, 0.9, -0.8\}$ ，且  $\delta = 2$  时，表 2 给出了对应的约束子集  $\Omega'_o$  中各个向量的具体取值。最后，将  $\Omega'_o$  的各个取值代入到式(11)中，实现对未知字段  $\mathbf{U}$  的估计。

由例子可得，子集约束的方法可以将原来大小为  $2^5$  的取值空间的求和降为大小为  $2^2$  的子集求和。倘若上述例子中的校验字段长度为 2 bit，那么 MAP, Robust CRC 以及 BF-CRC 3 种算法的各自的运算消耗分别为： $32(M + C)$ ， $5 \times 4M + 5C$ ， $4M + 3C$ 。由此可知 BF-CRC 算法中因所需运算操作得到减少，极大的降低了运算复杂度，提高算法可实现性与实用性。

### 2.3 子集约束算法性能与相关参数的关系

根据 2.2 节的分析可知，子集约束算法的性能受信噪比估计和参数  $\delta$  大小的影响。至于无线通信系统的信噪比估计问题一直以来都是备受大家关注的问题，当前也已经取得许多重要成果。文献 [15] 对当前较为经典的信噪比估计算法进行了归纳总结，比较分析了各自的优缺点，随后文献 [16] 则

表 2 子集空间  $\Omega'_o$  计算示意表格

测试向量	当 $l_o = 5, \delta = 2$ 时, $\Omega'_o$ 中所对应的具体向量的取值				
	1.2	0.2	-0.1	0.9	-0.8
$e_0=00$	1	1	0	1	0
$e_1=01$	1	1	1	1	0
$e_2=11$	1	0	1	1	0
$e_3=10$	1	0	0	1	0

提出了一种新型的基于相关向量机的信噪比估计算法，该方法具有估计范围大、估计精度高的特点。

相对于参数  $\delta$  的选取问题，无线信道的信噪比估计技术已经相当成熟，因此本节主要是对 BF-CRC 算法中测试向量长度  $\delta$  展开讨论。根据图 1 (a) 可知，为了使得子集概率求和能够较好的逼近原始空间的结果，首先要保证发送向量  $\mathbf{o}$  能够处在约束的子集内。于是则有  $l_{\text{err}} < \delta$ ，记接收向量  $\mathbf{y}_o$  中出错的比特位数为  $l_{\text{err}}$ 。由于算法中是通过置信度来定位不确定比特，因而往往不一定能恰好保证接收向量中置信度低的比特正好都是错误比特。此时需要寻求具有指导意义的上界，为此这里以高斯信道下 BPSK 信号的传输为例进行讨论。根据图 2 中 BPSK 接收信号幅度的概率曲线可知置信度较低的比特分为两种可能：(1) 该比特的信号受到干扰但并未发生极性翻转；(2) 该比特的信号受到干扰且发生极性翻转。但信号信噪比相对较高时，发生极性翻转的比特的接收信号幅度则以很大概率在接收门限附近，所以在可以近似认为发生翻转比特的信号幅度都小于发射信号幅度的一半，例如 BPSK 信号中当信噪比为 4 dB 时上述事件的发生概率为  $P = 0.9691$ 。于是，设信号幅度小于发射信号幅度一半的区间为  $A$ ，需设置的测试向量长度  $\delta$  则以很大的概率满足下列条件 (其中  $P(H_0) = P(H_1) = 0.5, P(A | H_0) = P(A | H_1)$  成立)，

$$\begin{aligned} \delta/l_o &\leq P(H_0)P(A | H_0) + P(H_1)P(A | H_1) \\ &= P(A | H_0) \end{aligned} \quad (13)$$

再将 BPSK 信号的误比特特性代入式(13)，可得  $\delta$  依概率的约束范围，尽管该约束条件并不能非常确切的给出测试长度  $\delta$  的取值，但其在一定程度上反映了测试长度与信噪比的变化规律，对测试长度的选取具有指导意义。

$$l_o P_{e_{\text{bpsk}}}(\gamma) \leq \delta \leq l_o [P_{e_{\text{bpsk}}}(\gamma/4) - P_{e_{\text{bpsk}}}(9/4\gamma)] \quad (14)$$

### 3 算法仿真与分析

**仿真 1** 本文通过设定一种协议格式并作为实

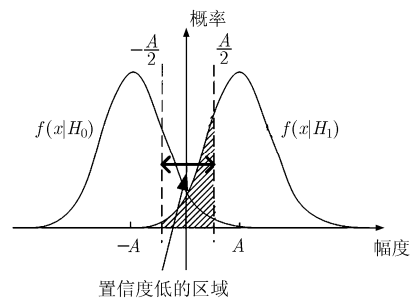


图 2 BPSK 信号的概率分布图

验对象来验证 BF-CRC 算法的有效性, 以 Matlab 2010b 为实验平台进行仿真, 最后将其性能与 Robust CRC 算法和 Min DIS 算法进行比较。设定协议中包括 8 bits 的固定字段  $K$ , 8 bit 的未知字段  $U$ , 30 bit 的不关心字段  $O$  以及利用 CRC8 的校验字段  $C$ 。设物理层传输的信号形式为 DBPSK, 并能提供置信度最低的  $\delta$  个比特的对应位置。虽然 Robust CRC 算法提到利用的软信息进行纠错, 但考虑到计算量等实际问题则实验中依旧采取硬判决信息进行处理与分析。在测试向量长度  $\delta$  取不同值条件下, 实验结果如图 3 所示。

由结果可知, 当设置的测试向量长度  $\delta$  较小时, BF-CRC 算法的纠错性能较差, 在信噪比为 7 dB 时其首部错误率为  $6.0 \times 10^{-5}$  逊色于 Robust CRC 算法的  $2.6 \times 10^{-5}$ , 同时可知 Robust CRC 算法和 Min DIS 算法在低信噪比的情况下纠错性能非常相近。当测试向量长度  $\delta$  取为 4 bit(或 6 bit)时, BF-CRC 算法尽管会在 -4 dB~2 dB 区间内纠错性能较差, 但随着信噪比的增加 BF-CRC 算法的性能越来越好, 甚至在信噪比在 5 dB 以上的高信噪比区间里 BF-CRC 算法性能超过 Robust CRC 算法。分析原因正如前面提到了在判断置信度最低的  $\delta$  个比特是由软信息引入的, 因而间接的利用了接收信号的软信息。于是相对于硬判决信息下的 Robust CRC 算法会稍有增益。由于实验中  $U$  字段长度较短, 所以对于 Min DIS 算法的纠错性能体现不明显。

在信道信噪比在 3~8 dB 范围内 BF-CRC 和 Robust CRC 两个算法的纠错性能都比 Min DIS 算法好, 这是由于校验字段中其实携有较多的有用信息。由前面分析可知如果算法在信噪比 4.8 dB 时取得较好果则理应满足条件  $0.41958 \leq \delta_{opt} \leq 6.5675$ , 由  $\delta = 4$  的曲线可验证 2.3 节的分析的正确性。

**仿真 2** 为了进一步说明 BF-CRC 算法的在具

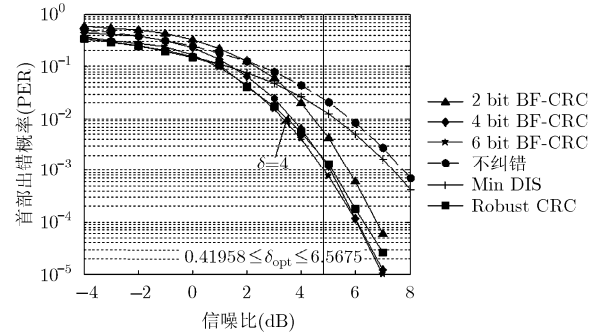


图 3 不同测试长度下 BF-CRC 与其他算法的性能对比

体情形中的实用性, 本文设置具体的实验场景如下: 无线接入点(Access Point, AP)发送的数据, 经过加性高斯白噪声信道(或慢衰落瑞利信道), 最后在用户端(User)接收。相对加性高斯白噪声信道而言, 在慢衰落瑞利信道条件下应用 BF-CRC 算法时需要增加考虑相位差异以及衰落因子等对置信度量造成的影响<sup>[17,18]</sup>。根据文献[4]中提到的协议压缩技术可知, 在该场景下 WIFI 的 802.11 MAC 和 IP 协议可以进行如下冗余分析<sup>[19]</sup>, 如图 4 所示。后续部分将详细解析无线终端同时开启多个下载业务的情形时, 网络协议中的冗余:

(1)对已经建立链接的下行数据, MAC 协议层中帧控制字段(Frame Control)、目的 MAC 地址(MAC Addr1), AP 的 MAC 地址(MAC Addr2)、源 MAC 地址(MAC Addr3)、序号控制(Seq)等字段都可以根据通信的前后数据帧以及下层协议推测得到。

(2)在 IP 协议层中, 实际应用时发现版本号 (Version)、服务类型(Service)、分片(Frag)和片偏移(Offset)等字段取值在通用的情形下是固定的, 属于字段  $K$ 。数据长度(TotalLen)字段其取值则可通过下层协议传输的服务数据单元长度得到, 属于可推

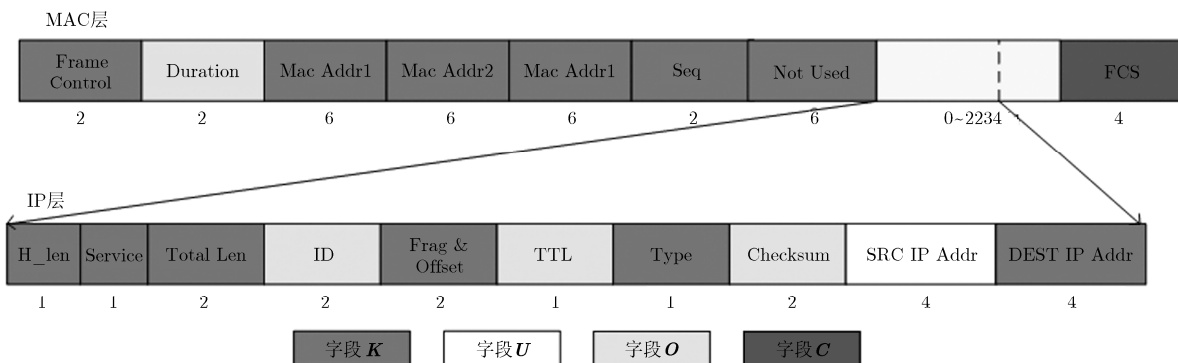


图 4 WIFI 的 802.11MAC 和 IP 协议格式

测字段  $P$ 。设下载业务都是使用相同的应用协议，因此可认为协议类型字段(Type)为已知。对于 IP 分组中的目的 IP 地址是在其接入网络时就已分配，也属于字段  $K$ 。

(3)其他一些字段取值虽不固定，例如标识字段(ID)、生存周期(TTL)和校验和(Checksum)，但不会影响到整个链接以及传输的正确性，因此定义为不关心字段  $O$  (根据校验的特性可知 Checksum 中其实也具有冗余信息，限于时间本文在此暂不予讨论)。

综上所述，对网络中数据首部纠错最终归结为对源 IP 地址的容错估计。在同时进行多个下载业务时，接收数据的源 IP 地址具有标识数据流的作用，因此根据 CRC 校验字段(FCS)对源端 IP 地址进行纠错具有十分重要的意义。

为了简便，特对 802.11 的无线 WIFI 信号进行简化，忽略扩频、信道编码等技术，并设信号的类型为 DBPSK 信号。在具体协议中，CRC 校验字段(FCS)长度为 4 Byte，因此正如前面举例提到的如果直接采用 Robust CRC 算法，则需要  $2^{32}$  个 double 类型的存储单元(大约为 16 GByte)来存储校验状态概率，如此的存储开销是难以承受的。因此文献[8]中提出次优的 Robust CRC 算法(sub-Robust CRC)，将 FCS 字段分成 4 个 8 bit 的被认为独立的校验字段，此时存储需求降为  $4 \times 2^8$  个 double 类型的存储单元(大约 4 kByte)，再利用 CRC 校验关系进行纠错。本部分主要将 BF-CRC 算法与可实现的 sub-Robust CRC 算法和 Min DIS 算法等进行性能对比。以 Microsoft Visual Studio 2008 为平台进行仿真实验。

由实验结果图 5 和图 6 可知，当测试向量长度

较短时，BF-CRC 算法的性能需要在高信噪比的情况下才具有效果；当测试长度相应的增加时，如取 6 bit，BF-CRC 算法纠错性能明显改善，对于加性高斯白噪声信道下在信噪比为 7 dB 时其协首部出错概率约为  $2 \times 10^{-5}$  而 sub-Robust CRC 算法的约为  $3 \times 10^{-4}$ ；对于慢衰落瑞利信道下信噪比 20 dB 时其协议首部出错率约为  $6 \times 10^{-6}$  明显优于 sub-Robust CRC 算法的  $2 \times 10^{-3}$ 。由前面描述知 sub-Robust CRC 算法中将 FCS 字段分解为 4 个 8 bit 近似独立的校验字段，如此便破坏了校验字段内部的约束关系造成性能的极大损失。因此当信噪比较高时，BF-CRC 算法纠错效果明显，且性能显著优于 sub-Robust CRC 算法。

#### 4 结束语

本文在提出了一种利用置信度信息的子集约束纠错算法(BF-CRC)，该算法利用每个比特的置信度信息构建以接收向量为中心的约束子集，降低计算联合后验概率时需遍历空间的大小，通过理论分析和实验验证得到 BF-CRC 算法具有算法复杂度低的特点。本文最后分别对抽象协议和 802.11 的无线 MAC 协议进行了仿真实验，仿真结果表明 BF-CRC 可通过调节测试长度保证较好的纠错性能，减少运算开销，增强系统可实现性，提高无线通信网络的抗噪声性能。值得注意的是 BF-CRC 仅仅只是利用了置信度信息来锁定不确定的比特的位置，倘若在计算似然概率时采用软信息的形式，其纠错性能将会进一步得到提高。再者，虽然本文只是重点讨论了利用 CRC 校验进行首部纠错，而其他协议层也有类似的校验关系，例如 IP 层的 Checksum 校验，因此对利用这类校验信息进行相应地纠错处理本文仍旧能给出有较好的借鉴效果。

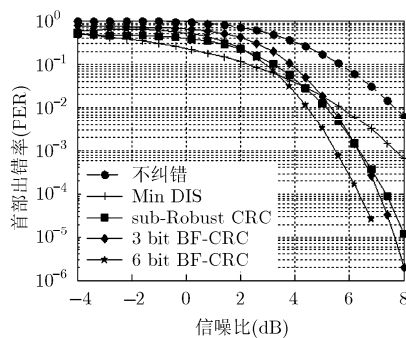


图 5 在 AWGN 信道下不同的纠错算法的性能对比

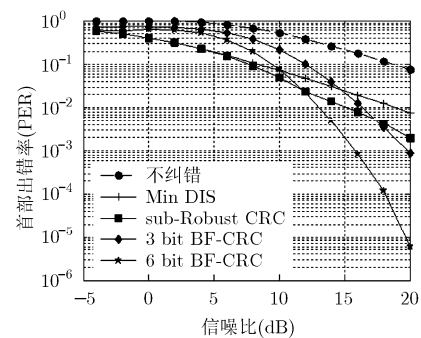


图 6 在 Rayleigh 信道下不同的纠错算法的性能对比

#### 参考文献

[1] Woo G R, Kheradpour P, Shen D, *et al.* Beyond the bits: cooperative packet recovery using physical layer information

[C]. Proceedings of the ACM Internet Conference on Mobile Computing and Network, Quebec, Canada, 2007: 147-158.

[2] Aman M N, Sikdar B, and Chan W K. Efficient packet

- recovery in wireless networks[C]. Proceedings of the Wireless Communications and Networking Conference (WCNC), Istanbul, Turkey, 2014: 1791-1796.
- [3] Wang S S, Sheu S T, Lee Y H, *et al.* CPR: a CRC-based packet recovery mechanism for wireless networks[C]. Proceedings of the Wireless Communications and Networking Conference (WCNC), Shanghai, China, 2013: 321-326.
- [4] Duhamel P and Kiffer M. Joint Source-channel Decoding: a Cross-layer Perspective with Applications in Video Broadcasting[M]. UK, Academic Press, 2009: 193-246.
- [5] 施里涛, 李欧, 王晓梅, 等. 一种高能效的无线传感器网络自主容错机制[J]. 电路与系统学报, 2013, 18(2): 102-107.
- Shi L T, Li O, Wang X M, *et al.* An active fault-tolerant scheme with high energy efficiency in wireless sensor networks[J]. *Journal of Circuits and Systems*, 2013, 18(2): 102-107.
- [6] Schmid F, Orlear D, and Wehrle K. A heuristic header error recovery scheme for RTP[C]. Proceedings of the Wireless On-demand Network Systems and Services (WONS), Alberta, Canada, 2013: 186-190.
- [7] Kiffer M and Duhamel P. Joint protocol and channel decoding: an overview[C]. Proceedings of the Future Network Mobile Summit, Florence, Italy, 2010: 1-16.
- [8] Marin C, Leprovost Y, and Kiffer M. Robust MAC-lite and soft header recovery for packetized multimedia transmission [J]. *IEEE Transactions on Communications*, 2010, 58(3): 775-782.
- [9] Meriaux F and Kiffer M. Robust IP and UDP-lite header recovery for packetized multimedia transmission[C]. Proceedings of the International Conference on Acoustics, Speech and Signal Processing(ICASSP), Texas, USA, 2010: 2358-2361.
- [10] Chase D. Class of algorithms for decoding block codes with channel measurement information[J]. *IEEE Transactions on Information Theory*, 1972, 18(1): 170-181.
- [11] 党小宇, 陶静, 虞湘宾, 等. 一种低复杂度的 Turbo 乘积码自适应 Chase 译码算法[J]. 电子与信息学报, 2014, 36(3): 739-743.
- Dang X Y, Tao J, Yu X B, *et al.* A low-complexity adaptive chase decoding algorithm for turbo product code[J]. *Journal of Electronics & Information Technology*, 2014, 36(3): 739-743.
- [12] Wolf J K. Efficient maximum likelihood decoding of linear block codes using a trellis[J]. *IEEE Transactions on Information Theory*, 1978, 24(1): 76-80.
- [13] Esmaeili M, Alampour A, and Gulliver T A. Decoding binary linear block codes[J]. *IEEE Transactions on Communications*, 2013, 61(6): 2138-2144.
- [14] Argon C and McLaughlin S W. An efficient chase decoder for turbo product codes[J]. *IEEE Transactions on Communications*, 2004, 52(6): 896-898.
- [15] 张金成, 彭华, 赵国庆. 信噪比估计算法研究[J]. 信息工程大学学报, 2011, 12(5): 535-542.
- Zhang J C, Peng H and Zhao G Q. Research on SNR estimation algorithm[J]. *Journal of Information Engineering University*, 2011, 12(5): 535-542.
- [16] 韩博, 吴杰, 许华, 等. 基于相关向量机的信噪比估计算法[J]. 通信学报, 2013, 34(4): 201-206.
- Han B, Wu J, Xu H, *et al.* New SNR estimation algorithm based on relevance vector machine[J]. *Journal on Communications*, 2013, 34(4): 201-206.
- [17] 冯战, 郑海昕, 秦铭晨. AWGN 与 Rayleigh 信道下 TPC 性能仿真研究[J]. 无线电工程, 2013, 43(9): 7-9.
- Feng Z, Zheng H X, and Qin M C. Performance simulation and research on turbo product codes over AWGN and Rayleigh channels[J]. *Ratio Engineering*, 2013, 43(9): 7-9.
- [18] 郑贺, 陆佩忠, 胡捍英. 基于二分图的乘积码迭代译码算法[J]. 电子与信息学报. 2006, 28(1): 86-90.
- Zheng H, Lu P Z, and Hu H Y. Iterative decoding algorithm for product codes based on bipartite graphs[J]. *Journal of Electronics & Information Technology*, 2006, 28(1): 86-90.
- [19] IEEE Std 802.11-2007. Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications[S]. 2007.
- 王晓梅: 女, 1976 年生, 博士, 讲师, 研究方向为无线自组织网络和网络数据协同处理.
- 范亮: 男, 1989 年生, 硕士生, 研究方向为网络数据协同处理.
- 陈彦: 男, 1990 年生, 硕士生, 研究方向为网络数据协同处理.