



系统工程与电子技术
Systems Engineering and Electronics
ISSN 1001-506X, CN 11-2422/TN

《系统工程与电子技术》网络首发论文

题目：基于两次窗口滑动运算的非等长帧同步字盲识别算法
作者：王原卿，呼鹏江，杨俊安，刘辉
收稿日期：2023-06-29
网络首发日期：2024-03-27
引用格式：王原卿，呼鹏江，杨俊安，刘辉. 基于两次窗口滑动运算的非等长帧同步字盲识别算法[J/OL]. 系统工程与电子技术.
<https://link.cnki.net/urlid/11.2422.tn.20240325.1243.008>



网络首发：在编辑部工作流程中，稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定，且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式（包括网络呈现版式）排版后的稿件，可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定；学术研究成果具有创新性、科学性和先进性，符合编辑部对刊文的录用要求，不存在学术不端行为及其他侵权行为；稿件内容应基本符合国家有关书刊编辑、出版的技术标准，正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性，录用定稿一经发布，不得修改论文题目、作者、机构名称和学术内容，只可基于编辑规范进行少量文字的修改。

出版确认：纸质期刊编辑部通过与《中国学术期刊（光盘版）》电子杂志社有限公司签约，在《中国学术期刊（网络版）》出版传播平台上创办与纸质期刊内容一致的网络版，以单篇或整期出版形式，在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊（网络版）》是国家新闻出版广电总局批准的网络连续型出版物（ISSN 2096-4188，CN 11-6037/Z），所以签约期刊的网络版上网络首发论文视为正式出版。

基于两次窗口滑动运算的非等长帧 同步字盲识别算法

王原卿, 呼鹏江, 杨俊安*, 刘辉
(国防科技大学电子对抗学院, 安徽 合肥 230037)

摘要: 在非合作通信场景下, 针对现有的非等长帧同步字盲识别算法存在的抗误码性能不佳、识别速度慢和识别结果不完整的问题, 本文提出了一种基于两次窗口滑动运算的非等长帧同步字盲识别算法。首先将比特流均匀切分为多个窗口, 取前两个窗口做滑动同或运算得到扩展同步字(extended synchronization word, E-SW); 然后利用得到的 E-SW 与剩余窗口分别做滑动相关性运算, 得到每个窗口对应的 E-SW 并组成 E-SW 集合; 最后对 E-SW 集合进行统计分析, 筛选出未知同步字的码字内容。仿真结果表明, 本文算法在误码率为 10^{-2} 时能达到 98.7% 的识别准确率, 并且可以识别出未知同步字的完整码字内容。在相同准确率的情况下, 本文算法比现有算法至少能多适应 0.01 的误码。此外, 本文算法的识别速度较现有算法更快, 且同样适用于等长帧协议。

关键词: 非合作通信; 同步字; 非等长帧; 盲识别; 窗口滑动
中图分类号: TN911.23 **文献标志码:** A

Blind synchronization word recognition algorithm for non-equal length frame based on two window-sliding operations

WANG Yuanqing, HU Pengjiang, YANG Junan*, LIU Hui
(Institute of Electronic Countermeasures, National University of Defense Technology, Hefei 230037, China)

Abstract: In non-cooperative communication scenarios, existing blind frame synchronization algorithms for non-equal length frame face significant challenges, including suboptimal error resilience, slow detection speed and incomplete recognition results. In order to solve these challenges, a blind recognition algorithm for non-equal length frame SYNC word based on two window-sliding operations is proposed. firstly, it divides the bitstream into multiple windows, and uses first two windows to obtain Extended Synchronization Word (E-SW) through sliding XNOR operation. Then, it uses sliding correlational operation between E-SW and the remaining windows to get the E-SWs for all windows and forms the E-SW set. Finally, statistical analysis is conducted on the E-SW set to determine the unknown SYNC word. The simulation results show that this algorithm has a recognition accuracy of 98.7% when the error rate is 10^{-2} , and recognize the complete codeword through unknown synchronous words. Under the same accuracy, this algorithm can adapt to at least 0.01 more-bit errors than existing algorithms. In addition, the speed of this algorithm is faster than existing algorithms and also applicable to equal length frame protocols.

Keywords: Non-cooperative communication; Synchronization word; Non-equal length frame; Blind identification; Window-sliding

0 引言

在现代通信系统中, 通信双方按照事先规定

好的通信协议进行通信, 但是在电子对抗、无线电资源管理等非合作通信场景下, 非合作方需要在通信协议未知的情况下对截获的比特流数据

进行分析。数据在数据链路层中以帧为单位进行封装并传输,所以为获取比特流数据中承载的有效信息,需要在协议类型未知的情况下对比特流数据进行正确的帧切分^[1]。为实现比特流数据的盲帧切分,首先需要解决同步字(synchronization word, SW)的盲识别问题。

目前国内外针对 SW 盲识别问题的研究对象大多是等长帧。对于等长帧可以借助 SW 在比特流中周期性出现的特性来进行盲识别^[2,3,4,5],但是这一特性在非等长帧中不存在,因此非等长帧的 SW 盲识别是一个十分具有挑战性的工作。随着通信体制的快速发展和大量未知私有协议的涌现,非等长帧协议已经在民用和军用领域被广泛投入使用^[6],例如 802.11 系列中的部分协议、CCSDS 中部分卫星通信协议和部分私有通信协议等,但是目前对于非等长帧 SW 盲识别的研究较少,且现有算法存在一定缺点和不足,因此研究一种有效的非等长帧同步字盲识别算法具有很重要的实际意义。

目前对于非等长帧的 SW 盲识别算法主要可以分为三大类:基于比特流码字特征的盲识别算法、基于数据挖掘的盲识别算法和基于神经网络的盲识别算法。

(1) 基于比特流码字特征的盲识别算法:文献[7]提出了一种基于软判决的同步字盲识别算法;文献[8]针对一类具有高码元密度特性的 SW,通过码元密度检测的方法实现数据链路层协议的帧同步盲识别,该算法的适用范围有限;文献[9]提出了基于最大公共子串的盲识别算法,但是该方法不适用于有误码的情况;文献[10]在已知 SW 长度和帧长范围的前提下利用帧结构遍历的方法进行盲识别,该方法属于半盲场景且算法复杂度较高;文献[11]利用多重分形谱实现 SW 盲识别,但是该方法无法确定 SW 的长度,所以其识别结果仅是 SW 的部分字段。

(2) 基于数据挖掘的盲识别算法:文献[12]通过有向图的形式构建频繁项之间的关联规则,实现同步字的准确唯一识别;文献[13]和[14]利用数据挖掘中的关联规则分析实现同步字的盲提取。但是基于数据挖掘的方法时间成本高且需要大量的数据支撑。

(3) 基于神经网络的盲识别算法:文献[15]利用卷积神经网络与线性滤波器结合实现同步字的准确提取;文献[16]利用循环神经网络

(recurrent neural network, RNN)具有记忆性的特性,利用两个循环神经网络实现了从非等长帧比特流中准确提取同步字和帧长。文献[17]和[18]利用深度学习技术进行了物理层帧同步的相关工作。但是基于神经网络的方法在训练阶段需要依赖标签数据,而在实际的应用场景下标签的获取是比较困难的,所以适用性不强,并且该方法需要预先训练神经网络参数因此算法训练耗时长。

此外,有学者通过引入其他先验信息辅助帧同步盲识别。文献[19-22]利用信道编码特性进行非等长帧协议的编码同步联合识别。文献[23]利用伪码特性进行同步。文献[24]和[25]通过引入信道状态进行帧同步联合识别。还有的学者利用时域信息来辅助码域的同步字识别^[26],但这些都不属于全盲识别的范畴。总的来看,现有的研究存在着抗误码性能不佳、识别速度慢和识别结果不完整等问题。

文献[2]利用一次窗口滑动的相关性运算实现等长帧的 SW 盲识别,但是该方法不适用于非等长帧问题,本文借鉴该方法中窗口滑动的思想,提出了一种基于两次窗口滑动运算的非等长帧同步字盲识别算法。本文提出的算法分为三步。首先将原始比特流数据均匀地切分为多个窗口,并取窗口一作为参考窗口与窗口二进行滑动同或运算,得到窗口一对应的 E-SW;其次将窗口一的 E-SW 作为新的参考窗口分别与剩余窗口做滑动相关性运算,得到每个窗口对应的 E-SW 并组成 E-SW 集合;最后对 E-SW 集合进行统计分析,筛选出准确的 SW 的码字内容。仿真结果验证了本文算法对非等长帧协议的有效性,在接收数据存在一定误码的情况下,本文算法比现有算法的识别准确率更高、速度更快。此外,本文算法同样适用于等长帧协议,并且识别准确率略优于文献[2]提出的等长帧算法。

1 系统模型

本文考虑的示例场景如图 1 所示^[3]。其中 A 和 B 分别是合作通信方的发射机和接收机,二者之间通过发送连续的非等长帧数据进行通信。C 是非合作通信方的侦察干扰机,其目标是干扰 AB 之间的有效通信。C 在窃听信道上侦听 AB 之间的通信内容,并在没有先验信息的基础上从比特流数据中准确识别出 AB 之间用于通信同

步的 SW，然后利用特定的干扰方式破坏 AB 之间通信的同步环节，使二者通信中断。

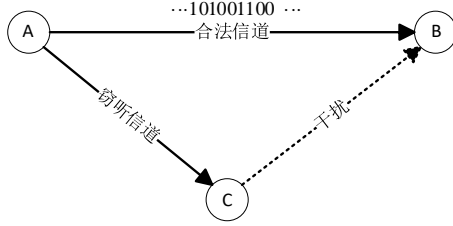


图1 本文的示例场景

Fig. 1 Example scenario in this article

如图2所示，发射机通过合法信道向接收机发送连续的非等长帧数据 $X = (x_1, x_2, \dots, x_i, \dots)$ ，其中 $x_i \in \{0, 1\}$ 。侦察机通过窃听信道截获长度为 M 的比特流数据 $Y = (y_1, y_2, \dots, y_M)$ 。非等长帧的结构由 SW 和数据字段 (Data) 两部分组成，在图中用深色部分表示 SW，浅色部分表示 Data。SW 在传输时保持不变，用 $S = (s_1, s_2, \dots, s_L)$ 来表示，其中 L 代表 SW 的长度。Data 的内容和长度会根据传输信息的不同而变化，所以每一帧的帧长是不同的。此外，由于 Data 部分往往会采取加密、加扰、交织等处理^[6]，所以 Data 通常表现为随机的 01 序列。

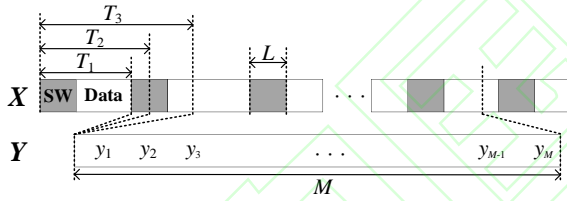


图2 发射数据、接收数据和非等长帧的结构

Fig. 2 The structure of transmitting data, receiving data, and non-equal length frames

因为非合作通信方无法准确得知合法信道上何时开始通信，所以侦察机接收到的数据仅是从原始数据流中截获的一部分，因此数据 Y 的起始位置存在三种情况：位于帧头的起始位置、位于 SW 中某个位置或位于 Data 中某个位置，这三种情况分别对应于图2中的 T_1 、 T_2 和 T_3 。

数据在传输的过程中会受到信道噪声的影响，导致比特流数据中出现误码，而非合作方无法采用有效的纠错措施减少这种影响，所以截获数据会存在一定的误码现象。本文考虑在误码场

景下对同步字进行盲识别，并使用二进制对称信道 (binary symmetric channel, BSC) 来模拟信道误码对数据的影响。

2 算法描述

图3展示了本文算法的基本流程。本文算法主要分为三步，如左半部分框图所示，而每一步的具体实现细节如右半部分框图所示。每一步的具体实现细节将在下文介绍。

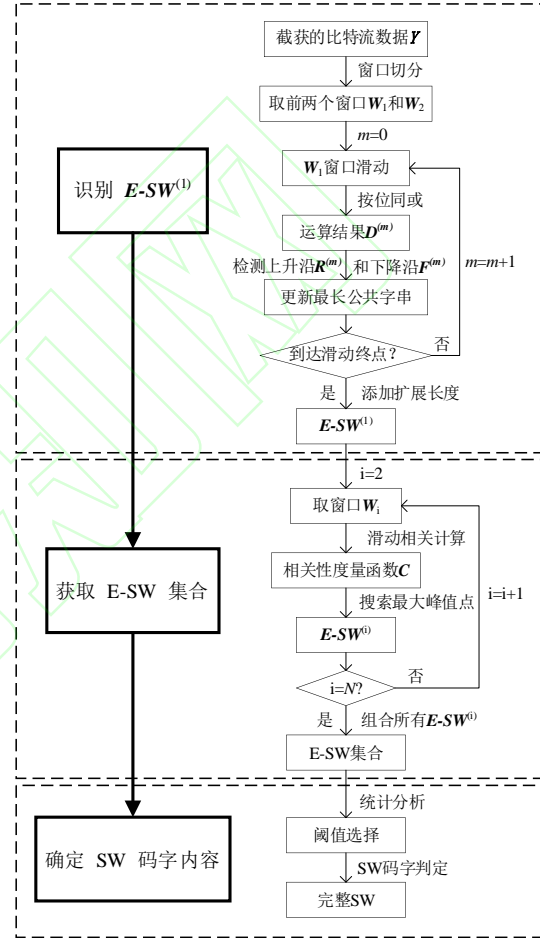


图3 本文算法的基本流程

Fig. 3 The basic process of the algorithm in this article

2.1 识别 $E-SW^{(1)}$

首先将截获到的比特流数据 Y 按长度 K 均匀切分，不足长度 K 的部分丢弃，如图4所示。

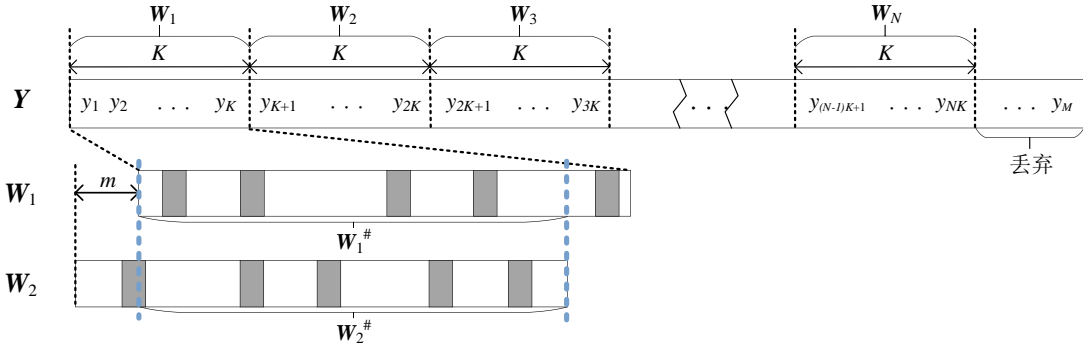


图4 比特流数据等长窗口切分和窗口滑动示意图

Fig. 4 Schematic diagram of equal-length window segmentation and window sliding for bitstream data

将切分得到的每段数据定义为窗口，一共可以得到 $N = \lfloor M / K \rfloor$ 个窗口，其中 $\lfloor \cdot \rfloor$ 表示向下取整运算，窗口集合用 $\{W_1, W_2, \dots, W_N\}$ 表示。 K 的取值是一个大于最大帧长的数，以保证每个窗口内至少存在一个完整的帧，在全盲场景下， K 可以设置为一个相对较大的数值。

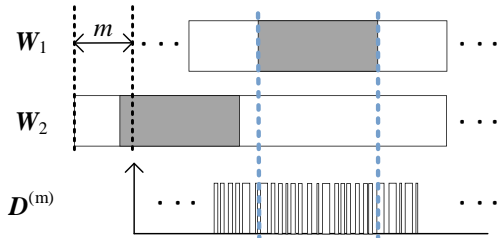
如图4所示，将参考窗口设置为第一个窗口 $W_1 = (y_1, y_2, \dots, y_K)$ ，与第二个窗口 W_2 做滑动同或运算， m 代表参考窗口滑动的位移。当参考窗口 W_1 滑动到 m 处时，对应的截断窗口为 $W_1^\# = (y_1, y_2, \dots, y_{K-m})$ ，其中 $m = 0, 1, \dots, K-1$ ， W_2 与之对齐的截断窗口为 $W_2^\# = (y_{K+1+m}, y_{K+2+m}, \dots, y_{2K})$ 。对 $W_1^\#$ 和 $W_2^\#$ 按位同或可得：

$$D^{(m)} = W_1^\# \odot W_2^\# \quad (1)$$

其中， \odot 代表同或运算。

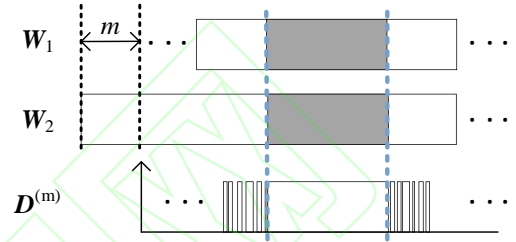
对于公式(1)得到的结果，如图5所示，在没有误码的情况下存在以下三种情况：

- (1) 两个窗口之间无 SW 对齐，表现为 $D^{(m)}$ 是杂乱无章的 01 序列，如图 5(a) 所示；
- (2) 两个窗口之间的 SW 对齐，表现为 $D^{(m)}$ 中存在一个较长连续为 1 的片段，如图 5(b) 所示；
- (3) 两个窗口之间的 SW 仅部分对齐或因随机性在数据部分出现部分相等，表现为 $D^{(m)}$ 可能存在多个较短连续为 1 的片段，并且连续的长度小于真实 SW 的长度 L ，如图 5(c) 所示。



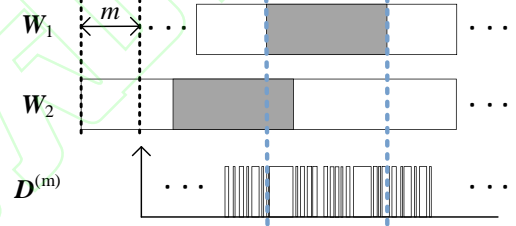
(a) 两窗口间无 SW 对齐

(a) No SW alignment between two windows



(b) 两窗口间 SW 对齐

(b) SW alignment between two windows



(c) 两窗口间部分 SW 对齐或数据部分存在相同
(c) Partial SW alignment or identical data between two windows

图5 滑动过程中存在的各种情况

Fig. 5 Various situations during the sliding process

对于上述3种情况，只有图5(b)对应的情况可以匹配到正确的 SW，所以可以通过检测序列 $D^{(m)}$ 中连续为 1 的长片段来判断 SW 的存在。本文使用上升沿和下降沿来检测连续为 1 的长片段。在实际情况下接收数据会存在误码，所以同步字可能会被 SW 中出现的误码所截断，导致出现多个连续为 1 的片段，如图 6 所示。针对此现象，本文对上升沿和下降沿进行重新定义：

$$\mathbf{R}^{(m)} = \left\{ \delta_r \left| \sum_{j=\delta_r-J}^{\delta_r-1} D^{(m)}(j) < \eta_1 \ \& \ \sum_{j=\delta_r}^{\delta_r+J-1} D^{(m)}(j) > \eta_2, \right. \right. \\ \left. \left. \delta_r \in [J+1, K-J+1] \right. \right\}$$

(2)

$$\mathbf{F}^{(m)} = \left\{ \delta_f \left| \begin{array}{l} \sum_{j=\delta_f-J+1}^{\delta_f} D^{(m)}(j) > \eta_2 \ \& \ \sum_{j=\delta_f+1}^{\delta_f+J} D^{(m)}(j) < \eta_1, \\ \delta_f \in [J, K-J] \end{array} \right. \right\} \quad (3)$$

其中, $\mathbf{R}^{(m)}$ 和 $\mathbf{F}^{(m)}$ 分别表示滑动位移为 m 时对应的上升沿和下降沿集合; δ_r 和 δ_f 分别表示上升沿和下降沿对应的位置坐标; J 代表可容纳错误的范围; η_1 和 η_2 分别为下降沿阈值和上升沿阈值。 η_1 一般选取略大于 $\lceil J/2 \rceil$ 的数, η_2 一般选取略小于 J 的数, $\lceil \cdot \rceil$ 代表向上取整运算。

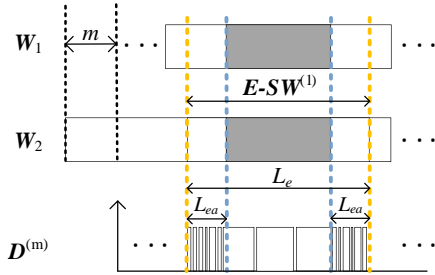


图6 存在误码情况下的 $D^{(m)}$ 取值情况

Fig. 6 The value of $D^{(m)}$ in the presence of error codes

这样的设置可以容许一定的误码,而且一般来说 \mathbf{SW} 的长度要大于几个字节,所以此设置不会对识别结果造成影响。将一对连续的上升沿和下降沿之间的码字 ($y_{\delta_r}, y_{\delta_r+1}, \dots, y_{\delta_f}$) 称为公共字符串,对应的坐标范围为 (δ_r, δ_f) 。在每个滑动位移 m 下,可能会得到多个公共字符串,取其中最长的作为最长公共字符串。参考窗口在滑动时,如果新得到的最长公共字符串比旧的要长,则更新最长公共字符串。将遍历所有 m 得到的最长公共字符串作为 \mathbf{SW} 候选项,对应的坐标范围为 (δ_r^*, δ_f^*) 。

由于比特流数据仅有 0 和 1 两种可能取值,所以在 \mathbf{SW} 候选项的边缘处可能会出现数据部分存在相同的现象,这会导致 \mathbf{SW} 候选项的长度大于真实 \mathbf{SW} 长度。此外,如果在边缘处出现了误码,则会导致 \mathbf{SW} 候选项的长度小于真实 \mathbf{SW} 长度。因此,坐标范围 (δ_r^*, δ_f^*) 并不一定是准确的 \mathbf{SW} 边界,即存在边缘模糊的现象。为了解决边缘模糊问题,定义了如图 6 所示的扩展长度 L_{ea} ,以保证 \mathbf{SW} 候选项中一定存在一个完整的 \mathbf{SW} 。在原坐标范围 (δ_r^*, δ_f^*) 的基础上,两边各扩展 L_{ea} bits 长度的码字,即坐标范围变为 $(\delta_r^* - L_{ea}, \delta_f^* + L_{ea})$ 。最终可以得到:

$$\mathbf{E-SW}^{(1)} = (e_1^{(1)}, e_2^{(1)}, \dots, e_{L_e}^{(1)}) \quad (4)$$

$$= (y_{\delta_r^* - L_{ea}}, \dots, y_{\delta_r^*}, \dots, y_{\delta_f^*}, \dots, y_{\delta_f^* + L_{ea}})$$

其中,将式(4)中的码字定义为扩展同步字 ($\mathbf{E-SW}$),其长度为 L_e 。为了后续阐述问题方便,用 $(e_1^{(1)}, e_2^{(1)}, \dots, e_{L_e}^{(1)})$ 来表示 $\mathbf{E-SW}^{(1)}$ 内的码字内容。

2.2 获取 E-SW 集合

将 $\mathbf{E-SW}^{(1)}$ 作为新的参考窗口,滑动匹配其余窗口得到其对应的 $\mathbf{E-SW}^{(i)}$,以组成 $\mathbf{E-SW}$ 集合。这一步可以等价为合作方的帧同步过程^[27,28],如图 7 所示,将 $\mathbf{E-SW}^{(1)}$ 作为已知同步字,与窗口 \mathbf{W}_i 进行滑动匹配,匹配得到 \mathbf{W}_i 中相关程度最高的长度为 L_e 的码字,并将其作为窗口 \mathbf{W}_i 对应的 $\mathbf{E-SW}^{(i)}$,其中滑动位移为 n 。

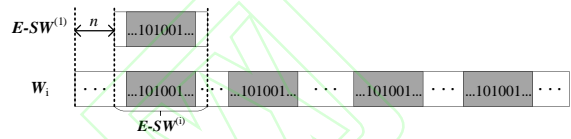


图7 $\mathbf{E-SW}^{(1)}$ 与窗口 \mathbf{W}_i 滑动匹配得到其对应的 $\mathbf{E-SW}^{(i)}$

Fig. 7 Sliding match with $\mathbf{E-SW}^{(1)}$ and \mathbf{W}_i to obtain corresponding $\mathbf{E-SW}^{(i)}$

对于 $0 \leq n \leq K - L_e$, 结合二进制比特流的特性定义相关性度量函数的公式^[29]:

$$C(\mathbf{W}_i, n) = \frac{1}{L_e} \sum_{t=1}^{L_e} e_t^{(1)} \odot w_{t+n}^{(i)}, n \in [0, K - L_e] \quad (5)$$

其中, $\mathbf{W}_i = (w_1^{(i)}, w_2^{(i)}, \dots, w_K^{(i)})$ 。

对于窗口 \mathbf{W}_i , 遍历所有 n 可以得到关于 n 的相关性度量函数的曲线。由于 \mathbf{SW} 在比特流中固定不变,所以当两个窗口中的 \mathbf{SW} 位置对齐时相关性度量函数 $C(\mathbf{W}_i, n)$ 会出现一个峰值。若 \mathbf{W}_i 内存在多个 \mathbf{SW} ,则会得到多个峰值。相关性度量函数曲线的示例仿真图像如图 8 所示,仿真采用的同步字为随机生成的长度为 32 bits 的码字“1101100111011010011110111101010”, Data 长度范围为 40-80 bits,窗口切割长度 $K = 500$ bits,扩展长度 $L_{ea} = 8$ bits。

本文选取相关性度量函数的最大峰值点对应的滑动位移 n^* 作为 $\mathbf{E-SW}^{(i)}$ 的起始位置,截取长度为 L_e 的码字部分作为窗口 \mathbf{W}_i 对应的 $\mathbf{E-SW}^{(i)}$,记作:

$$\mathbf{E-SW}^{(i)} = (w_{n^*+1}^{(i)}, w_{n^*+2}^{(i)}, \dots, w_{n^*+L_e}^{(i)}) \quad (6)$$

$$= (e_1^{(i)}, e_2^{(i)}, \dots, e_{L_e}^{(i)})$$

对每个窗口进行上述滑动相关性计算,可以得到每个窗口对应的扩展同步字。最后将所有窗口的 $\mathbf{E-SW}^{(i)}$ 组成 $\mathbf{E-SW}$ 集合:

$\{E - SW^{(1)}, E - SW^{(2)}, \dots, E - SW^{(N)}\}$ 。

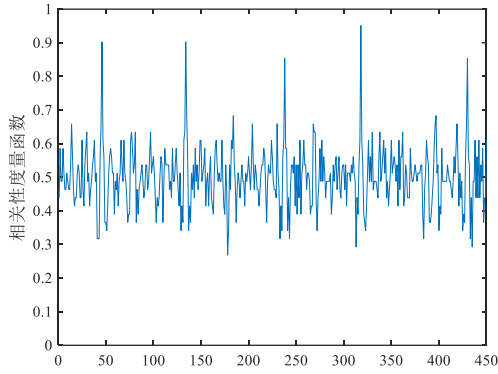


图 8 相关性度量函数曲线仿真图

Fig. 8 Simulation diagram of correlation measurement function curve

2.3 确定 SW 码字内容

对于 E-SW 集合中的任一 $E - SW^{(i)}$ ，其存在固定域和可变域。固定域对应于 SW 部分，在无误码的情况下取值固定不变；可变域是扩展长度区域，其每一位的取值是不固定的，即为随机的 0 或 1。在存在误码的情况下，固定域内的部分比特位会出现错误，但是从统计的角度来看，每一比特位的取值仍然是大多数相同，这与可变域比特位取值的统计特性存在明显的区别。所以根据固定域和可变域的统计性质，可以对 E-SW 集合中的每一比特位进行统计分析。

将 E-SW 集合中的所有 $E - SW^{(i)}$ 放入矩阵 γ 中，得到：

$$\gamma = \begin{bmatrix} E - SW^{(1)} \\ E - SW^{(2)} \\ \vdots \\ E - SW^{(N)} \end{bmatrix} = \begin{bmatrix} e_1^{(1)} & e_2^{(1)} & \dots & e_{L_e}^{(1)} \\ e_1^{(2)} & e_2^{(2)} & \dots & e_{L_e}^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ e_1^{(N)} & e_2^{(N)} & \dots & e_{L_e}^{(N)} \end{bmatrix} \quad (7)$$

对 γ 的每一列进行求和并归一化，得到：

$$\gamma_{sum} = \frac{1}{N} \begin{bmatrix} \sum_{i=1}^N e_1^{(i)} & \sum_{i=1}^N e_2^{(i)} & \dots & \sum_{i=1}^N e_j^{(i)} & \dots & \sum_{i=1}^N e_{L_e}^{(i)} \end{bmatrix} = [r_1 \quad r_2 \quad \dots \quad r_j \quad \dots \quad r_{L_e}] \quad (8)$$

根据 γ_{sum} 分析固定域和可变域。在无误码的情况下，可变域的取值接近于 0.5，而固定域的取值为 0 或 1。但是由于存在误码的影响，固定域内的若干码字会出现错误，导致固定域的 γ_{sum} 取值接近但不等于 0 或 1，此时就需要合适的阈值来判断某位置是否为固定域并确定其码字内容。

首先分析 E-SW 集合中的每一个扩展同步字的取值。对于 $E - SW^{(i)}$ ，其每个比特位的取值仅有 0 和 1 两种选择并且相互独立，所以每个比特位的取值服从二项分布。令 τ 为 BSC 信道的转移概率，按每个比特位的取值将其分为三种情况：

- (1) 可变域：服从参数为 0.5 的二项分布；
- (2) 固定域取值为 1：取值为 1 的概率为 $1-\tau$ ，所以服从参数为 $1-\tau$ 的二项分布；
- (3) 固定域取值为 0：取值为 1 的概率为 τ ，所以服从参数为 τ 的二项分布。

接下来分析统计结果 γ_{sum} 的取值。棣莫弗-拉普拉斯中心极限定理指出：当试验次数足够大时，二项分布以正态分布为其极限分布。所以当 N 取值较大时， γ_{sum} 中每一位置的取值服从相应的正态分布，且根据正态分布的 3 方差准则，每个位置的取值可以认为分布在对应的 3 个区间内（分别对应于前述 3 种情况）：

- (1) 可变域：取值近似服从均值为 0.5、方差为 $0.25/N$ 的正态分布，对应的三倍标准差为 $1.5/\sqrt{N}$ ，根据正态分布的 3 方差准则，其取值可以认为分布在区间一 $[0.5 - 1.5/\sqrt{N}, 0.5 + 1.5/\sqrt{N}]$ 内。
- (2) 固定域取值为 1：同理，取值分布在区间二 $[1 - \tau - 3\sqrt{(1-\tau)\tau/N}, 1 - \tau + 3\sqrt{(1-\tau)\tau/N}]$ 内；
- (3) 固定域取值为 0：同理，取值分布在区间三 $[\tau - 3\sqrt{(1-\tau)\tau/N}, \tau + 3\sqrt{(1-\tau)\tau/N}]$ 内。

当 $0 \leq \tau < 0.5$ 且 $N > 0$ 时，区间一的下界大于区间三的上界，区间一的上界小于区间二的下界。所以可以使用两个阈值 μ_1 和 μ_2 来区分上述三种情况，其中 $\mu_1 > \mu_2$ 。

为保证可以容忍一定的误差，我们选取区间一上界与区间二下界的中点作为阈值 μ_1 ，区间三上界与区间一下界的中点作为阈值 μ_2 ，阈值的计算公式如下：

$$\mu_1 = \frac{3}{4} - \frac{\tau}{2} + \frac{1}{\sqrt{N}} \left(\frac{3}{4} - \frac{3}{2} \sqrt{(1-\tau)\tau} \right) \quad (9)$$

$$\mu_2 = \frac{1}{4} + \frac{\tau}{2} - \frac{1}{\sqrt{N}} \left(\frac{3}{4} - \frac{3}{2} \sqrt{(1-\tau)\tau} \right) \quad (10)$$

最后，利用阈值 μ_1 和 μ_2 对 γ_{sum} 中的每一项进行判断：

- 若 $\mu_2 \leq r_j \leq \mu_1$ ，则判断其为非 SW 部分，并

- 且丢弃;
- 若 $\mu_1 < r_j$, 则判断其为 SW 部分且取值为 1;
 - 若 $r_j < \mu_2$, 则判断其为 SW 部分且取值为 0。
- 遍历 γ_{sum} 中所有的项即可得到完整的 SW。

2.4 计算复杂度分析

算法第一步是两个长度为 K 的窗口间的滑动同或运算, 其计算复杂度为 $O(K^2)$; 算法第二步需要用长度为 L_e 的 E-SW 对剩下的 $N-2$ 个窗口分别进行相关性计算, 计算复杂度为 $O((N-2)KL_e)$; 算法第三步仅需要对矩阵的每一列进行求和, 复杂度为 $O(L_e)$ 。由此可见, 算法的总复杂度主要取决于前两步, 所以总复杂度可以表示为 $O(K^2 + (N-2)KL_e)$ 。并且本文算法面向二进制数据, 涉及到的主要运算皆可转化为加法运算, 因此识别速度较快。

3 仿真实验

本文使用识别准确率和计算量作为评价指标, 其中 SW 识别正确的标准为 SW 长度识别正确且 SW 的每一位码字也识别正确。实验的参数设置如表 1 所示。本文提出的算法是为任意 SW 而设计的, 所以每次蒙特卡洛实验时采用随机生成的二进制序列作为 SW。

本文在配置为 Core-i7-12700KF 处理器、3.6GHz 主频、16GB 内存和 Windows11 操作系统的计算机上使用 Matlab 软件进行仿真实验。

表 2 算法各中间步骤的输出结果

Table 2 Output results of each intermediate step of the algorithm

变量	输出结果	长度或维度
$E-SW^{(1)}$	1011110100100100001111001111000111110011100111000 1011110100100100001111001111000111110011100111000 101011010010010000111001111100111110011100010001 01001111001001000011100111100011111001101101100 110100000010010000111001111000111110011111000011 1010101000100100001110011110001101100110101010 000101110010010000111001111000111110011110111001	49 bits N 维*49 bits (N=63)
γ_{sum} (保留 2 位小数)	[0.59 0.54 0.48 0.60 0.46 0.49 0.44 0.54 0 0 0.98 0.03 0.02 1 0 0 0 0.05 1 1 0.98 0 0.02 1 0.97 1 1 0.03 0 0.02 1 1 1 1 0.98 0.98 0 0 0.98 1 0.52 0.51 0.44 0.46 0.52 0.57 0.46 0.62 0.54]	49 bits
SW	00100100001110011110001111110011	32 bits

扩展同步字中都包含了完整的 SW 并且 SW 在各扩展同步字中对应的比特位置也都相同, 这证明利用相关性度量来匹配 E-SW 是有效的。

(3) 从 γ_{sum} 的结果可以看出, 在固定的 SW 域 (粗斜体码字部分) 的取值接近于 0 或 1, 而在扩展域部分的取值接近于 0.5, 这与前述理论分析结果一致, 证明利用统计分析的方式进行

表 1 实验参数设置 Table 1 Experimental parameter settings	
参数	数值或设置方式
同步字长度/bits	32
数据字段长度范围/bits	400-800
帧数	200
误码率变化范围	0-0.1
扩展长度 L_{ea} /bits	8
切割窗口长度 K /bits	2000
接收数据起始位置	随机设置
蒙特卡洛实验次数	1000

3.1 算法性能分析

3.1.1 识别结果分析

在误码率为 1% 的条件下, 以随机生成的 “00100100001110011110001111110011” 为同步字进行单次仿真实验, 算法各步骤的中间输出结果如表 2 所示。用粗斜体表示各输出码字中的 SW 部分, 用下划线表示识别结果中存在的误码情况。通过对表中结果的分析可知:

(1) 从 $E-SW^{(1)}$ 的识别结果中可以看出, $E-SW^{(1)}$ 中包含了完整的 SW, 并且码长也符合本文的扩展长度设置, 同时注意到, $E-SW^{(1)}$ 中存在误码现象 (被下划线标记的码字), 这说明本文定义的上升沿和下降沿可以容忍一定的误码并能正确地找到 SW。

(2) 分析 E-SW 集合的结果可以看出, 各

SW 筛选是可行的。

(4) 算法最终得到的 SW 与真实的 SW 一致, 说明了本文算法可以在误码场景下准确地识别出未知同步字的码字内容。

3.1.2 阈值的选择及有效性分析

本文 2.3 中给出了不同误码率下最佳判定阈值的计算式(9)和(10), 根据上述公式计算得到的

最佳阈值曲线如图 9 中两条实线所示。在全盲场景下,非合作方往往无法预先得知接收数据的误码率信息,所以需要提前确定阈值。本文将最佳阈值曲线的均值作为固定阈值,如图 9 中两条虚线所示,其中固定阈值 1 和固定阈值 2 的取值分别为 0.7863 和 0.2137。

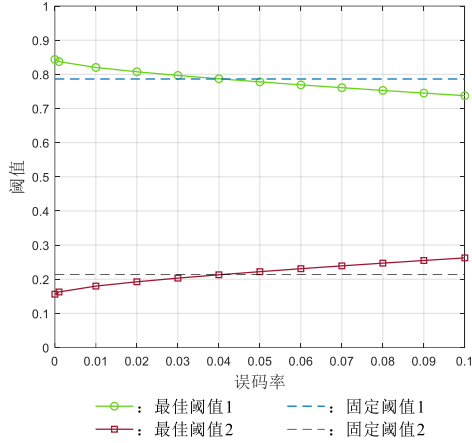


图 9 最优阈值曲线和固定阈值曲线

Fig. 9 The optimal threshold and fixed threshold curve

为验证阈值对识别准确率的影响以及证明固定阈值选择的有效性,实验选取了不同阈值进行比较,仿真结果如图 10 所示。从图中可以看出,计算得到的最佳阈值具有最佳的准确率性能,而固定阈值的识别准确率十分接近最佳阈值,所以可以证明固定阈值选择的合理性和有效性,因此在之后的实验中沿用此阈值设置。

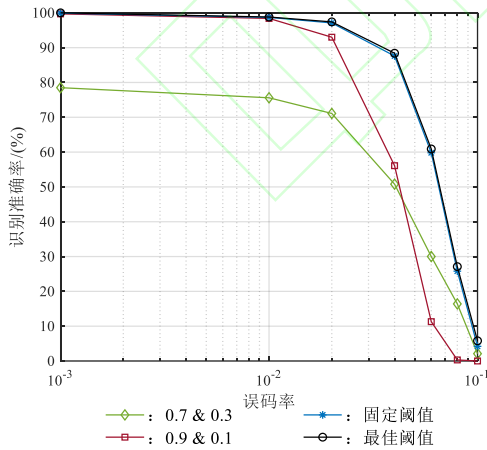


图 10 不同阈值条件下的性能对比

Fig. 10 Comparison under different threshold condition

3.1.3 关键参数分析

本节将讨论窗口长度 K 和扩展长度 L_{ea} 对识别准确率的影响。在总数据量不变的基础上,将

K 分别设置为 1000、2000 和 3000,将 L_{ea} 分别设置为 4、8 和 12,其余参数保持不变,实验结果如图 11 所示。

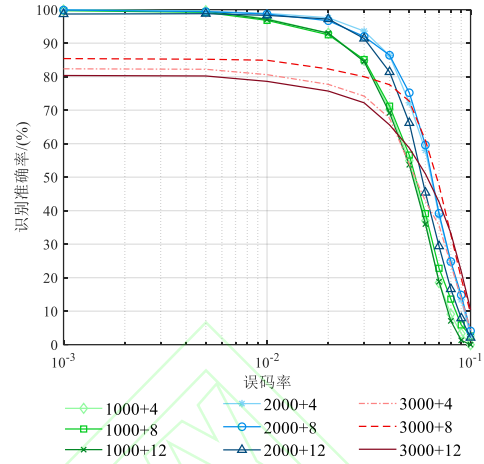


图 11 不同窗口长度和扩展长度下的识别准确率对比

Fig. 11 Comparison of recognition accuracy under different window length and extension length

首先从理论上分析算法产生错误的主要原因,当算法第一步的 E-SW 中没有包含完整的同步字或算法第三步中同步字的判定出现错误时,算法的输出结果会出现错误。从图中可以看出,窗口长度和扩展长度对算法的影响相对独立。

固定扩展长度,可以看出窗口长度 K 对算法识别准确率的影响较大。当 K 较小时,单个窗口中包含的完整 SW 就较少,所以受误码影响算法第一步可能出现错误,因而导致识别准确率有略微下降;当 K 较大时,虽然算法第一步中的 E-SW 识别更加准确了,但由于总数据量不变,所以窗口的数量 N 会随之减少,而算法第三步中的阈值计算是基于大数定理推导的,所以当 N 减少时,阈值选取的准确性下降,使得识别准确率下降程度较大,但是这一问题可以通过增加数据量来解决。综合分析,选取 $K = 2000$ 的窗口作为本文使用的参数。

固定窗口长度,可以看出扩展长度 L_{ea} 对算法识别准确率的影响较小。当 L_{ea} 较小时,可能会使 E-SW 无法包含完整的 SW,而当 L_{ea} 较大时,则会增加算法第三步中的判决错误的概率。所以综合考虑后选取 $L_{ea} = 8$ 作为本文使用的扩展长度。

3.2 算法性能对比

3.2.1 非等长帧情况下的算法性能对比

实验选取文献[16]、文献[10]和文献[11]的算

法与本文算法进行性能对比。文献[16]算法在传统方法的基础上利用深度学习技术来辅助识别，使用的是 RNN 网络；文献[10]算法在已知帧长范围和同步字长范围的前提下，利用帧结构遍历的方式识别出 SW 的码字内容，其中同步字长范围的参数设置为 30-35 bits，迭代次数为 5 次；文献[11]算法利用多重分形谱原理对信息字段进行删减，然后通过计算剩余序列中的字段浓度筛选出 SW，删减行数和删减次数分别设置为 20 行和 50 次。文献[16]和文献[10]中算法的输出结果是完整的 SW，但是文献[11]中算法的输出仅是 SW 的部分字段，所以在本节中将找到 SW 的部分字段作为文献[11]算法识别正确的标准。

在识别准确率方面，仿真结果如图 12 所示。在无误码的情况下，4 种算法的识别准确率都接近 100%，但是随着误码率的升高，算法的识别准确率随之下降。本文算法在误码率为 1% 时的准确率为 98.7%，当误码率升高至 5% 时仍然具有 75% 以上的准确率。文献[10]提出的算法在误码率为 2% 时，准确率已不足 50%，而误码率高于 4% 后基本上无法正确识别出 SW。文献[16]和[11]提出的算法识别准确率较高，但仍不及本文算法。在误码率较低时，准确率的差距小于 5%，但是随着误码率的升高差距开始变大，当误码率大于 5% 时，差距超过了 20%。虽然文献[16]和[11]算法的识别准确率与本文算法的差距较小，但是文献[16]算法需要借助海量的计算资源，而文献[11]算法仅能找到 SW 的部分码字。

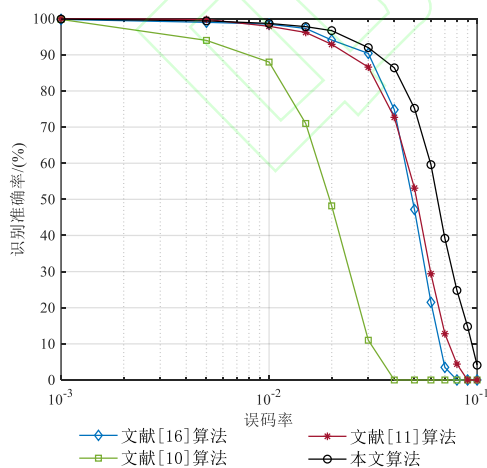


图 12 本文算法与现有算法的识别准确率对比

Fig. 12 Comparison of recognition accuracy between this algorithm and existing algorithms

在算法的计算量方面，由于文献[16]借助了

深度学习的方法，所以计算量较大，故本文不对比该算法的计算量。将误码率设置为 1%，每种算法分别统计 100 次蒙特卡洛实验的平均计算量，结果如表 3 所示。文献[10]算法的计算量大约是本文算法的 10 倍，文献[11]虽然与本文算法的总计算量相近，但是该算法的主要运算为乘法运算，对于计算机而言，计算乘法消耗的时间是加法的 4 倍以上，所以综合分析本文算法总计算量要显著小于现有算法，证明了本文算法具有更好的时效性。

表 3 算法计算量比较

Table 3 Comparison of Calculation Quantity			
算法	文献[10]	文献[11]	本文算法
加法计算/次	74022032	10704	7830187
乘法计算/次	5020	6018801	49

接下来对不同算法的实验结果进行讨论。文献[16]利用神经网络强大的特征学习能力可以有效处理 SW 盲识别问题，但是随着误码率升高，比特流中的 SW 特征会变得模糊，所以识别效果变差；文献[10]虽然利用大数定理减少误码的影响，但是在遍历数据时并没有采取抗误码措施，所以在统计分析前的 SW 检测步骤会出现检测错误的可能，导致识别准确率不佳，此外该算法需要在每种 SW 长度和帧长范围内对所有数据进行完全遍历，因此计算成本很高；文献[11]在误码率较高时，由于 SW 内出现较多误码，导致了原有的数学规则被破坏，因此在误码率较高的时候其识别准确率不高，此外该算法需要将二进制数据转化为十进制数据，并进行大量的乘法运算，所以算法计算量偏高；本文提出的算法通过改进上升沿下降沿判定方式、增加扩展长度、对 E-SW 集合进行统计分析等容错措施，减少了误码对识别准确率的影响，所以抗误码性能较好，此外本文算法虽然需要进行两次窗口滑动运算，但是每次运算只需在单个窗口内滑动，而不需要每次都遍历全部数据，因此算法的识别速度更快。综上所述，本文提出的算法与现有算法相比识别准确率更高、算法速度更快并且可以识别出完整的 SW。

3.2.2 等长帧情况下的算法性能对比

等长帧是非等长帧的一种特例，所以本文算法从理论上同样适用于等长帧协议。为验证本文算法对等长帧协议的有效性，选取仅适用于等长帧协议的文献[2]和[5]中的算法进行性能比较。

将数据字段的长度固定为 800 比特, SW 沿用之前的设置, 仿真结果如图 13 所示。

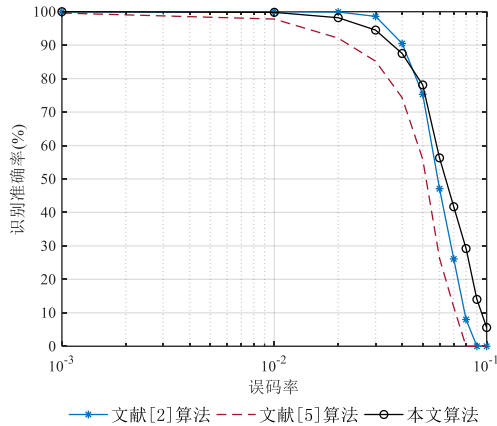


图 13 本文算法与文献[2]算法在等长帧数据上的识别准确率对比

Fig. 13 Comparison of recognition accuracy between the algorithm in this paper and the algorithm in reference[2] on equal length frame data

对于等长帧数据, 本文算法较文献[5]算法的识别准确率更高。与文献[2]算法相比, 两者的识别准确率相近, 当误码率为 4% 时, 识别准确率均在 90% 左右。当误码率小于 4% 时, 本文算法的识别准确率略低于文献[2]算法, 这是因为本文算法不借助 SW 周期性出现的性质, 所以识别准确率略低; 但是当误码率较高时 (大于 4% 时), 本文算法的识别准确率要优于文献[2]算法, 这是因为文献[2]算法仅采用一次窗口滑动的操作, 而本文算法采用两次窗口滑动操作, 其可以保证在高误码率的情况下减少因误码导致的 SW 匹配错误。上述仿真结果证明了本文算法在等长帧数据上的适用性。

3.3 算法在现有非等长帧协议上的性能表现

实验选取了 802.11b 协议、CCSDS-TC 协议、某导弹数据链协议和某私有电台协议四种实际非等长协议帧作为数据输入, 并在生成仿真数据时严格按照协议的标准文档。采用已知协议测试

表 4 各非等长帧协议的 SW 参数

Table 4 SW parameters of each non-equal length frame protocol

协议帧	同步字 (十六进制表示)	同步字长/bits
802.11b 协议	FFFFFFFFFFFFFFFFFFFFFFFF3A0	144
CCSDS-TC 协议	C5C5C5C5C5C579EB90	80
某导弹制导数据链协议	0163D0FF214FAAE0C5665FBCDDCA944B 4673C6C22EBDBE1A6B6A09D9260748E2	256
某私有电台协议	69966996966996	64

是为了检验算法的有效性, 并不借助协议的已知信息, 各协议帧的 SW 参数如表 4 所示。实验中的 SW 设定为在每一帧帧头部分中固定不变的码字, 其可能是 SW、前导码和帧定界符的组合 [30]。一般协议的数据字段变化范围较大, 所以将数据字段的长度范围扩大至 [800, 4000]。

本文算法在各非等长帧协议上的识别准确率如图 14 所示, 可以看出本文算法对实际非等长协议都具有较高的识别准确率, 证明了本文算法具有一定的实际应用价值。CCSDS-TC 协议、某导弹数据链协议和某私有电台协议的识别准确率较高, 但是 802.11b 协议的识别准确率较上述 3 种协议低。通过分析 802.11b 协议的同步字可以看出, 其存在 132 个重复为 1 的码字片段, 这可能会导致算法第二步出现匹配位置不精准的错误, 其在最终的 SW 识别结果中表现为: 大部分的码字识别正确而在边缘处的少量码字出现错误, 从而导致了识别准确率略低。

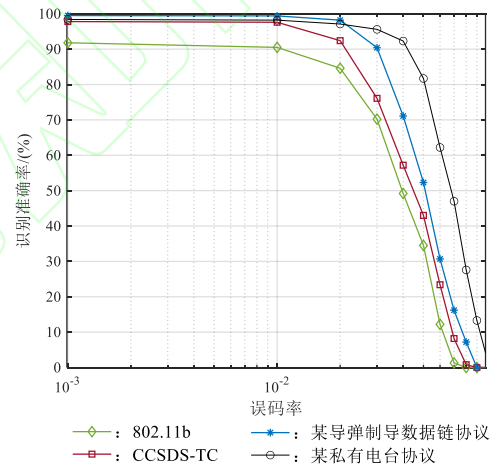


图 14 本文算法在各非等长帧协议上的识别准确率

Fig. 14 The recognition accuracy of this algorithm on various non-equal length frame protocols

4 结论

在非合作通信场景下, 同步字盲识别是分析

非等长帧协议的基础,目前非等长帧协议的同步字盲识别算法仍存在抗误码性能较差、识别速度慢和识别结果不完整等问题。为解决上述问题,本文提出了一种基于两次窗口滑动运算的非等长帧同步字盲识别算法。算法主要包括窗口间的滑动同或运算、滑动相关性运算和统计分析三个关键步骤,其可以在接收数据存在误码的情况下识别出完整的非等长帧同步字。仿真结果表明,在误码场景下,本文算法在识别准确率和识别速度两个方面皆优于现有算法,并且该算法同样适用于等长帧协议。未来将进一步研究小数据量下的非等长帧同步字盲识别问题,以降低算法对数据量的需求。

参考文献

- [1] 薛开平, 柳彬, 王劲松, 等. 面向链路比特流的未知帧关联分析[J]. 电子与信息学报, 2017, 39(2): 374-380.
XUE K P, LIU B, WANG J S, et al. Data link bit stream oriented association on unknown frame[J]. Journal of Electronics & Information Technology, 2017, 39(2): 374-380.
- [2] KIL Y S, LEE H, KIM S H, et al. Analysis of blind frame recognition and synchronization based on Sync word periodicity[J]. IEEE Access, 2020, 8: 147516-147532.
- [3] PRATIK P B, KALYANKUMAR B. Non-cooperative denial of communication after synchronizing with repeating sequences[C]//Proc. of the Defense Science Research Conference and Expo, 2011.
- [4] XU Y Y, ZHONG Y, HUANG Z P. An Improved blind recognition algorithm of frame parameters based on Self-Correlation[J]. Information, 2019, 10(2): 64-72.
- [5] 白戔, 杨晓静, 王懋. 基于相关滤波和哈达玛变换的帧同步码识别[J]. 探测与控制学报, 2011, 33(3): 69-72, 80.
BAI Y, YANG X J, WANG M. Recognition method of frame synchronization codes based on relativity filter and Hadamard transformation algorithm[J]. Journal of Detection and Control, 2011, 33(3): 69-72, 80.
- [6] SUWANSANTISUK W, CHIANI M, WIN M Z. Frame synchronization for variable-length packets[J]. IEEE Journal on Selected Areas in Communications, 2008, 26(1): 52-69.
- [7] QIN J Y, HUANG Z P, LIU C W, et al. Novel blind recognition algorithm of frame synchronization words based on Soft-Decision in digital communication Systems[J]. PLoS ONE, 2015, 10(7): e0132114.
- [8] 熊颢, 雷迎科, 吴子龙. 基于码元密度检测的帧同步码盲识别算法[J]. 探测与控制学报, 2021, 43(1): 73-78.
XIONG H, LEI Y K, WU Z L. Frame synchronization code blind recognition based on symbol density detection[J]. Journal of Detection & Control, 2021, 43(1): 73-78.
- [9] 陈庆超, 王韬, 冯文博, 等. 基于最长公共子串挖掘的未知链路层协议帧切割算法[J]. 计算机科学, 2020, 47(7): 227-230.
CHEN Q C, WANG T, FENG W B, et al. Unknown link layer protocol frame segmentation algorithm based on longest common substrings mining[J]. Computer Science, 2020, 47(7): 227-230.
- [10] 李永辉, 陈小莉, 吴彩凤, 等. 一种非等长帧同步识别与参数提取算法[J]. 电子测量技术, 2019, 42(22): 123-128.
LI Y H, CHEN X L, WU C X, et al. A non-equal length frame algorithm of synchronization recognition and parameter extraction[J]. Electronic Measurement Technology, 2019, 42(22): 123-128.
- [11] 李歆昊, 张旻, 韩树楠. 基于多重分形谱的链路层协议帧同步字盲识别[J]. 电子与信息学报, 2017, 39(7): 1666-1672.
LI X H, ZHANG M, HAN S N. Frame synchronization word identification of link layer protocol based on multi-fractal spectrum[J]. Journal of Electronics & Information Technology, 2017, 39(7): 1666-1672.

- [12] LEI Y K , CAO C H. Frame segmentation in the link layer bit stream data based on directed graph[C]//Proc. of the 12th International Conference on Communication Software and Networks, 2020: 58-62.
- [13] LI X H, MA T , QIAN Q S. Frame synchronization method based on association rules for CNAV-2 messages[J]. Chinese Journal of Electronics, 2023, 32(2): 295-302.
- [14] PAN C S, GUO Z P, YANG L, et al. Frame synchronization word identification of link layer protocol based on N-ECLAT algorithm[C]//Proc. of the 8th International Conference on Instrumentation & Measurement, Computer, Communication and Control, 2018.
- [15] SONG J M, KIL Y S, KIM S H. Blind frame Syncword detection using deep neural networks with input linear filtering[C]//Proc. of the International Conference on Information and Communication Technology Convergence , 2019: 1039-1041.
- [16] KIL Y S, SONG J M, KIM S H, et al. Deep learning aided blind synchronization word estimation[J]. IEEE Access, 2021, 9: 30321-30334.
- [17] YANG T C, HE D X, LU Z P, et al. BiLSTM-based frame synchronization for overlapped S-AIS signals: A Learning Empowered Approach[C]//Proc. of the IEEE/CIC International Conference on Communications in China, 2023.
- [18] SARUNAS K, LOUISE H C, ROBERT W S. Training deep filters for Physical-Layer frame synchronization[J]. IEEE Open Journal of the Communications Society, 2022, 3: 1063-1075.
- [19] RODRIGUE I, SEBASTIEN H. On blind frame synchronization of LDPC codes[J]. IEEE Communications Letters, 2021, 25(10): 3190-3194.
- [20] DING X H, ZHOU K X, LI G Y, et al. Customized joint blind frame synchronization and decoding methods for analog LDPC decoder[J]. IEEE Trans. on Communications, 2023, 1-15.
- [21] DING Y, HUANG Z P, LI L Q. Joint blind frame synchronization and encoder identification of LDPC codes[C]//Proc. of the 7th International Conference on Intelligent Computing and Signal Processing, Xi'an, China, 2022: 825-829.
- [22] FENG Z X, LIU Y, ZHANG S Y, et al. Polar-coding-assisted blind frame synchronization based on soft information of frozen bits[J]. IEEE Communications Letters, 2023, 27(10): 2563-2567.
- [23] 华博, 毛忠阳, 康家方, 等. 基于非等量采样的伪码同步技术研究[J]. 系统工程与电子技术, 2022, 44(4): 1401-1408.
- HUA B, MAO Z Y, KANG J F, et al. Research on pseudo-random noise code synchronization technology based on non-commensurate sampling[J]. Systems Engineering and Electronics, 2022, 44(4): 1401-1408.
- [24] ANTONIO A D and MICHELE M. Symbol-Spaced feedforward techniques for blind bit synchronization and channel estimation in FSO-OOK communications[J]. IEEE Trans. on Communications, 2024, 72(1): 361-374.
- [25] SAMEER B, AMEER P M, and DAVID K R. Synchronization techniques for underwater acoustic communications[J]. International Journal of Communication Systems, 2023, 36(15): 1-22.
- [26] JIN M C, ZHANG S T, HE X D, et al. Blind recognition of frame synchronization in Time-Code domain[C]//Proc. of the 2023 International Conference on Ubiquitous Communication, 2023.
- [27] RODRIGUE I, SEBASTIEN H. Theoretical analysis of a MAP based blind frame synchronizer[J]. IEEE Trans. on wireless communications, 2009, 8(11): 5472-5476.
- [28] LIANG Y S, DINESH R, OREN E E. Sequential frame synchronization based on

- hypothesis testing with unknown channel state information[J]. IEEE Trans. on Communications, 2018, 63(8): 2972-2984.
- [29] SUI H T. Optimum frame synchronization and performance over binary symmetric channel[J]. Journal of Electronics, 1992, 9(3): 200-208.
- [30] RAMAMOORTHY M S, JALIHAL D, RAMAIYAN V. Optimal frame synchronization under general arrivals[J]. IEEE Trans. on Communications, 2018, 66(11): 5704-5717.

作者简介

王原卿 (2000-), 男, 硕士研究生, 主要研究方向为通信对抗。

呼鹏江 (1990-), 男, 讲师, 博士, 主要研究方向为通信对抗。

杨俊安 (1965-), 男, 教授, 博士, 主要研究方向为信号处理、智能对抗。

刘辉 (1983-), 男, 讲师, 博士, 主要研究方向为通信对抗、智能信息处理。