多重周期序列联合二次复杂度的计算

董丽华，胡予濮，曾勇
（西安电子科技大学 计算机网络与信息安全教育部重点实验室，陕西 西安 710071）

摘 要：以 Rizomiliotis 所提出的计算单序列的二次复杂度算法为基础，结合线性方程组的解的判定方法，给出了一个求解任意有限域上多重周期序列联合二次复杂度的算法。算法的复杂性分析表明算法复杂度至多为序列长度的三次函数。

关键词：密码学；流密码；二次复杂度；多重序列
中图分类号：TP309 **文献标识码**：A **文章编号**：1000-436X(2012)06-0011-08

Computing on the joint quadratic complexity of multiple periodic sequences

DONG Li-hua, HU Yu-pu, ZENG Yong
(Key Laboratory of Computer Networks and Information Security,
Ministry of Education School of Computer Science & Technology, Xidian University, Xi'an 710071, China)

Abstract: An algorithm for determining the joint quadratic complexity of the prescribed multiple periodic sequences over any finite field was presented by using the algorithm for computing the quadratic complexity of the prescribed single sequence proposed by Rizomiliotis and the methods for determining the solutions of the linear equations. The total processing time requirement is cubics function of the sequence length at most.

Key words: cryptography; stream cipher; quadratic complexity; multiple sequences

1 引言

二次复杂度是评估流密码的密钥流伪随机性的一个基本度量指标。自文献[1]中引入二次复杂度的概念之后，国外密码学界对其进行了大量的相关研究[2~6]。但是由于其处理的困难性，直到 2005 年 Rizomiliotis[7]才给出了一个计算二次复杂度的有效算法，然而该算法主要是针对单序列的。出于有效性考虑，向量化流密码在现实生活中尤其是需要实时数据传输的应用中的重要性已日益凸显。而目前有关多序列复杂度测度的研究结果主要集中于联合线性复杂度[8~17]。对于联合二次复杂度，可以注意到这样一个事实：对于如下的 7 周期 3 重序列，利用已有算法可以计算得到其联合线性复杂度为 7，各分量序列的二次复杂度分别为 1、2 和 3，而在第 6 节中将会看到此多重序列的联合二次复杂度只为 4。

$$\begin{pmatrix} 1\,0\,1\,0\,1\,0\,1\cdots \\ 0\,1\,0\,0\,0\,0\,0\cdots \\ 0\,1\,1\,1\,0\,0\,1\cdots \end{pmatrix}$$

我们既不能由多重序列的各分量序列的二

收稿日期：2010-08-23；**修回日期**：2010-11-08
基金项目：国家自然科学基金重点基金资助项目（60833008）；国家自然青年科学基金资助项目（61100235）；陕西省自然科学基础研究计划基金资助项目（2009JQ8005）；高等学校创新引智基地基金资助项目(B08038)
Foundation Items: The State Key Program of National Natural Science of China (60833008); The National Natural Science Funds of China for Young Scholar (61100235); The Natural Science Basic Research Plan in Shaanxi Province of China (2009JQ8005); 111 Project of China (B08038)

次复杂度也不能由其联合线性复杂度来确切地确定其联合二次复杂度，由此看到联合二次复杂度的研究与联合线性复杂度的研究具有同样的重要性，为此在本文中以 Rizomiliotis 算法为基础，利用线性方程组的解的判定方法对多重周期序列的联合二次复杂度进行了研究，设计了一个求解任意有限域上多重周期序列联合二次复杂度的算法，并给出了一个算法的应用实例。

章节安排如下：在第 2 节中对用于生成预定的多重序列的最小长度二次反馈移位寄存器(FSR)的综合问题进行了描述。在第 3 节中给出了算法的理论基础。第 4 节描述了计算联合二次复杂度的算法。第 5 节对算法的复杂度做了简要分析。第 6 节给出了一个算法的应用实例。第 7 节是结束语。

2 问题描述

本节给出了文中要用到的定义与记号及问题描述，所考虑的序列为任意有限域上的周期序列。

设 t 和 N 是 2 个正整数。所谓的 t 维 N 周期多重序列 s 定义如下：

$$s=\begin{pmatrix} s_1 \\ \vdots \\ s_j \\ \vdots \\ s_t \end{pmatrix}=\begin{pmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,N} \\ \vdots & \vdots & & \vdots \\ s_{j,1} & s_{j,2} & \cdots & s_{j,N} \\ \vdots & \vdots & & \vdots \\ s_{t,1} & s_{t,2} & \cdots & s_{t,N} \end{pmatrix}$$

其中，不失一般性地假设多重周期序列 s 的各分量序列 $s_j(1\leqslant j\leqslant t)$，具有公共地周期 N（不一定为最小周期），同时 $s_{j,i}$ 为第 j 个分量序列 s_j 的第 i 个元素，$1\leqslant i\leqslant N$，$1\leqslant j\leqslant t$。

生成给定的多重周期序列 s 的二次 FSR 的综合问题可以描述如下：

以 $f(x)=a_0+a_1x_1+a_2x_2+a_{1,2}x_1x_2+\cdots+a_mx_m+a_{1,m}x_1x_m+\cdots+a_{m-1,m}x_{m-1}x_m$，$a_0=1$，作为反馈函数的 m 状态二次 FSR 生成给定的 t 维 N 周期多重序列 $\boldsymbol{s}$ 的充要条件为

$$f(s_{j,i},s_{j,i+1},\cdots,s_{j,i+m-1})=s_{j,i+m},\ 1\leqslant i\leqslant N-m,\ 1\leqslant j\leqslant t \tag{1}$$

定义 生成给定的 t 维 N 周期多重序列 s 的最短二次 FSR 的长度称为多重序列 s 的联合二次复杂度，记为 $q^{(t)}(N)$。

由式(1)显然给定的 t 维 N 周期多重序列 s 的联合二次复杂度的计算等价于搜索使得式(2)成立的最小 m 值。

$$\boldsymbol{M}^{(t)}(N,m)\boldsymbol{F}(m)=\boldsymbol{E}^{(t)}(N,m) \tag{2}$$

其中，$\boldsymbol{F}(m)$为由待求解的二次 FSR 的反馈函数 $f(x)$的系数所构成的$[1+m(m+1)/2]\times 1$ 矩阵，即 $\boldsymbol{F}(m)=(a_0\ a_1\ a_2\ a_{1,2}\cdots a_m\ a_{m-1,m}\cdots\ a_{1,m})^{\mathrm{T}}$，同时由式(1)知对应的线性方程组(2)的系数矩阵 $\boldsymbol{M}^{(t)}(N,m)$ 为如下的 $[t(N-m)]\times[1+m(m+1)/2]$ 矩阵，而 $[1+m(m+1)/2]\times 1$ 矩阵 $\boldsymbol{E}^{(t)}(N,m)=(s_{1,m+1}\ s_{2,m+1}\ \cdots\ s_{t,m+1}\ s_{1,m+2}\ s_{2,m+2}\ \cdots\ s_{t,m+2}\ \cdots\ s_{1,N}\ s_{2,N}\ \cdots\ s_{t,N})^{\mathrm{T}}$。

$$\begin{pmatrix}
1 & s_{1,1} & s_{1,2} & s_{1,1}s_{1,2} & \cdots & s_{1,m} & s_{1,m-1}s_{1,m} & \cdots & s_{1,1}s_{1,m} \\
1 & s_{2,1} & s_{2,2} & s_{2,1}s_{2,2} & \cdots & s_{2,m} & s_{2,m-1}s_{2,m} & \cdots & s_{2,1}s_{2,m} \\
\cdots & & & & & & & & \\
1 & s_{t,1} & s_{t,2} & s_{t,1}s_{t,2} & \cdots & s_{t,m} & s_{t,m-1}s_{t,m} & \cdots & s_{t,1}s_{t,m} \\
1 & s_{1,2} & s_{1,3} & s_{1,2}s_{1,3} & \cdots & s_{1,m+1} & s_{1,m}s_{1,m+1} & \cdots & s_{1,2}s_{1,m+1} \\
1 & s_{2,2} & s_{2,3} & s_{2,2}s_{2,3} & \cdots & s_{2,m+1} & s_{2,m}s_{2,m+1} & \cdots & s_{2,2}s_{2,m+1} \\
 & \cdots & & & & & & & \\
1 & s_{t,2} & s_{t,3} & s_{t,2}s_{t,3} & \cdots & s_{t,m+1} & s_{t,m}s_{t,m+1} & \cdots & s_{t,2}s_{t,m+1} \\
 & & & & & & & & \\
 & \cdots & & & & & & & \\
1 & s_{1,N-m} & s_{1,N-m+1} & s_{1,N-m}s_{1,N-m+1} & \cdots & s_{1,N-1} & s_{1,N-2}s_{1,N-1} & \cdots & s_{1,N-m}s_{1,N-1} \\
1 & s_{2,N-m} & s_{2,N-m+1} & s_{2,N-m}s_{2,N-m+1} & \cdots & s_{2,N-1} & s_{2,N-2}s_{2,N-1} & \cdots & s_{2,N-m}s_{2,N-1} \\
\cdots & & & & & & & & \\
1 & s_{t,N-m} & s_{t,N-m+1} & s_{t,N-m}s_{t,N-m+1} & \cdots & s_{t,N-1} & s_{t,N-2}s_{t,N-1} & \cdots & s_{t,N-m}s_{t,N-1}
\end{pmatrix}$$

显然，此时线性方程组(2)有解的充要条件为

$$\mathrm{rank}(\boldsymbol{M}^{(t)}(N,m))=\mathrm{rank}(\boldsymbol{M}^{(t)}(N,m)|\boldsymbol{E}^{(t)}(N,m)) \tag{3}$$

因而 t 维 N 周期多重序列 s 的联合二次复杂度的计算等价于搜索使得式(3)成立的最小 m 值。

定理 1 设 $m^{(j)}(N)$为第 j 个 N 周期分量序列 s_j 的二次复杂度，则给定的 h 个序列 $s_k(k=1,2,\cdots,h)$的联合二次复杂度 $q^{(h)}(N)$一定不小于任意 $h-1$ 个分量序列的联合二次复杂度 $q^{(h-1)}(N)$，也不会小于这 h 个分量

序列的二次复杂度的最大值，即

$q^{(h)}(N)\geqslant q^{(h-1)}(N)$且 $q^{(h)}(N)\geqslant\max\{m^{(k)}(N)|1\leqslant k\leqslant h\}$

证明　首先，给定的 h 个序列 $s_k(k=1,2,\cdots,h)$的联合二次复杂度 $q^{(h)}(N)$ 是能够同时生成这 h 个分量序列的二次 FSR 的最小长度，因而此二次 FSR 必然也可以同时生成任意的 $h-1$ 个分量序列。

其次，这 $h-1$ 个分量序列的联合二次复杂度是所有能够同时生成这 $h-1$ 个分量序列的二次 FSR 的最短长度。因而，给定的 h 个序列 $s_k(k=1,2,\cdots,h)$，的联合二次复杂度 $q^{(h)}(N)$一定不小于任意 $h-1$ 个分量序列的联合二次复杂度 $q^{(h-1)}(N)$。

类似可证明 $q^{(h)}(N)\geqslant\max\{m^{(k)}(N)|1\leqslant k\leqslant h\}$。□

结论 1[7]　设 $q^{(h)}(n)=q^{(h)}(n-1)+\delta$，$\delta>0$，则对任意的 $i\in[0,\delta]$，等式 $q^{(h)}(n+i)=q^{(h)}(n)$成立。

注 1　下面第 4 节中将要给出的算法的思想主要基于定理 1 及结论 1，即首先计算第一个分量序列 s_1 的二次复杂度 $m^{(1)}(N)=q^{(1)}(N)$，随后根据定理 1 可知最初 2 个分量序列的联合二次复杂度 $q^{(2)}(N)$的最小可能值为 $q^{(1)}(N)$。因而可以由 $q^{(1)}(N)$开始搜索满足式(3)的最小值。在获得了最初的 k 个分量序列的联合二次复杂度 $q^{(k)}(N)$之后，可以由 $q^{(k)}(N)$开始计算最初的 $k+1$ 个分量序列的联合二次复杂度 $q^{(k+1)}(N)$。在算法的执行过程中可以利用结论 1 来进一步缩减算法的复杂度。□

注 2　然而，这里仍然有一个需要注意的问题，即如何确保最终获得的二次 FSR 能同时生成这所有的 h 个序列，在第 3 节中给出问题的答案。　□

结论 2　由定理 1，可以将矩阵 $\boldsymbol{M}^{(t)}(N,q^{(t)}(N))$排列如下：

$$\boldsymbol{M}^{(t)}(N,q^{(t)}(N))=(B_0(N,q^{(t)}(N))|B_1(N,q^{(t)}(N))|\cdots|B_{q^{(t)}(N)}(N,q^{(t)}(N)))$$

其中，$B_0(N,q^{(t)}(N))$为 $\sum_{j=1}^{t}(N-q^{(j)}(N))$ 维的全 1 列向量。对于 $q^{(k)}(N)<i\leqslant q^{(k+1)}(N)$，且 $i=q^{(k)}(N)+j$, $1\leqslant k<t$,

$$B_i(N,q^{(t)}(N))=\begin{pmatrix} s_{1,i} & s_{1,i-1}s_{1,i} & \cdots & s_{1,1}s_{1,i} \\ s_{1,i+1} & s_{1,i}s_{1,i+1} & \cdots & s_{1,2}s_{1,i+1} \\ & \cdots\cdots\cdots & & \\ X_{1,N+i-1-q_1} & X_{1,N+i-2+q_1}X_{1,N+i-1+q_1} & \cdots & s_{1,N-q_1}X_{1,N+i-1+q_1} \\ & \cdots\cdots\cdots & & \\ & \cdots\cdots\cdots & & \\ s_{t,i} & s_{t,i-1}s_{t,i} & \cdots & s_{t,1}s_{t,i} \\ s_{t,i+1} & s_{t,i}s_{t,i+1} & \cdots & s_{t,2}s_{t,i+1} \\ & \cdots\cdots\cdots & & \\ s_{t,N+i-1-q_t} & s_{t,N+i-2-q_t}s_{t,N+i-1-q_t} & \cdots & s_{t,N-q_t}s_{t,N+i-1-q_t} \end{pmatrix}$$

其中，以 q_j 记 $q^{(j)}(N)$，$1\leqslant j\leqslant t$，$X_{h,i}$ 由所需满足的相关关系式进行任意确定。对应的矩阵 $\boldsymbol{E}^{(t)}(N,q^{(t)}(N))$可以重新排列如下：

$$(s_{1,q_t+1}\ \ s_{1,q_t+2}\ \cdots\ X_{1,N+q_{t,1}}\ \ s_{2,q_t+1}\ \ s_{2,q_t+2}\cdots X_{2,N+q_{t,2}}\ \cdots\ s_{t,q_t+1}\ \ s_{t,q_t+2}\cdots\ s_{t,N})^{\mathrm{T}}\quad\square$$

注 3　由定理 1 知，$q^{(j)}(N)\geqslant m^{(j)}(N)$，$1\leqslant j\leqslant t$，因而

$$\sum_{j=1}^{t}(N-q^{(j)}(N))\leqslant 1+m(m+1)/2$$

注 4　在下文中，以 $s^{(h,n)}$ 记第 h 个分量序列 s_h 的最初 n 个元素；以 $q^{(h)}(n)$记 $s^{(h,n)}$与最初的 $h-1$ 个 N 周期分量序列的联合二次复杂度，相应的系数矩阵记为 $\boldsymbol{M}^{(h)}(n,q^{(h)}(n))$；矩阵 $\boldsymbol{M}^{(h)}(n,q^{(h)}(n))$的第 j 行（或第 j 列）记为 $R_j(n,q^{(h)}(n))$ (或 $C_j(n,q^{(h)}(n))$)。整数 j 称为行（或列）的下标。设

$$j=p(b,k)=b(b-1)/2+k \tag{4}$$

其中，$1\leqslant k\leqslant b\leqslant q^{(h)}(n)$，则列 $C_j(n,q^{(h)}(n))$是分块 $B_b(n,q^{(h)}(n))$的第 k 列。式(4)的逆运算如下：

$$(p_1^{-1}(j)\ \ p_2^{-1}(j))=(b\ k) \tag{5}$$

其中，

$$p_1^{-1}(j)=\left\lfloor\frac{1}{2}\sqrt{8j-7}+\frac{1}{2}\right\rfloor$$

$$p_2^{-1}(j)=j-\frac{1}{2}p_1^{-1}(j)(p_1^{-1}(j))-1$$

3　数学基础

本节将利用线性代数中线性方程组的解的判定方法回答注 2 中提出的问题。

引理 1[1]　设 $Mx=b$ 是具有解向量 a 的线性方程组。若对此方程组添加一个不以 a 为解向量的方程 $c^Tx=d$，即 $c^Ta\neq d$，则扩展方程组没有解的充要条件为

$$\mathrm{rank}\left(\begin{bmatrix}M\\ c^T\end{bmatrix}\right)=\mathrm{rank}(M)$$

定理 2　设有一个 m 长的二次 FSR 能同时生成最初的 $h-1$ 个分量序列，但是不能同时生成第 h 个分量序列 s_h 的前 $m+1$ 个元素 $s^{(h,m+1)}$，则没有 m 长的二次 FSR 能同时生成最初的 $h-1$ 个分量序列以及 $s^{(h,m+1)}$，即 $q^{(h)}(m+1)\neq m$ 的充要条件为

$\mathrm{rank}(\boldsymbol{M}^{(h)}(m+1,m))=\mathrm{rank}(\boldsymbol{M}^{(h-1)}(N,m)),2\leqslant h\leqslant t$

证明 由该 m 长的二次 FSR 能同时生成最初的 $h-1$ 个分量序列，知道线性方程组 $\boldsymbol{M}^{(h-1)}(N,m)\boldsymbol{F}(m)=\boldsymbol{E}^{(h-1)}(N,m)$有一个解，不妨设为 a。同时由该二次 FSR 不能生成 $s^{(h,m+1)}$，可知 a 不是方程 $f(s_{h,1},s_{h,2},\cdots,s_{h,m})=s_{h,1+m}$ 的解。

由引理 1 可知线性方程组 $\boldsymbol{M}^{(h)}(m+1,m)x=\boldsymbol{E}^{(h)}(m+1,m)$没有解，即没有 m 长的二次 FSR 能同时生成最初的 $h-1$ 个分量序列以及 $s^{(h,m+1)}$的充要条件为

$$\mathrm{rank}(\boldsymbol{M}^{(h)}(m+1,m))=\mathrm{rank}(\boldsymbol{M}^{(h-1)}(N,m)) \qquad \square$$

推论 1 设有一个 m 长的二次 FSR 能同时生成最初的 $h-1$ 个分量序列以及 $s^{(h,n)}$，但是不能生成 $s_{h,n+1}$，则没有 m 长的二次 FSR 能同时生成最初的 $h-1$ 个分量序列以及 $s^{(h,n+1)}$，即 $q^{(h)}(n+1)\neq m$ 的充要条件为

$$\mathrm{rank}(\boldsymbol{M}^{(h)}(n+1,m))=\mathrm{rank}(\boldsymbol{M}^{(h)}(n,m))$$

定理 3 若 $\mathrm{rank}(\boldsymbol{M}^{(h)}(n,m))\neq\mathrm{rank}(\boldsymbol{M}^{(h)}(n,m)|\boldsymbol{E}^{(h)}(n,m))$，则没有长度 d 小于 m 的二次 FSR 能同时生成最初的 $h-1$ 个分量序列以及 $s^{(h,n)}$，即 $q^{(h)}(n)\geqslant m$，其中 $q^{(h-1)}(N)\leqslant n\leqslant N$。

证明 由 $\mathrm{rank}(\boldsymbol{M}^{(h)}(n,m))\neq\mathrm{rank}(\boldsymbol{M}^{(h)}(n,m)|\boldsymbol{E}^{(h)}(n,m))$，知列 $\boldsymbol{E}^{(h)}(n,m)$与矩阵 $\boldsymbol{M}^{(h)}(n,m)$的各列线性独立。若选用的二次 FSR 的长度 $d<m$，则 $\sum_{j=1}^{h-1}(N-q^{(j)}(N))+n-m$ 维的向量 $\boldsymbol{E}^{(h)}(n,m)$ 仅是 $\sum_{j=1}^{h-1}(N-q^{(j)}(N))+n-d$ 维向量 $\boldsymbol{E}^{(h)}(n,d)$ 的一个子集。因而由线性代数的理论知，$\boldsymbol{E}^{(h)}(n,d)$与矩阵 $\boldsymbol{M}^{(h)}(n,d)$中的各列线性独立，即

$$\mathrm{rank}(\boldsymbol{M}^{(h)}(n,d))\neq\mathrm{rank}(\boldsymbol{M}^{(h)}(n,d)|\boldsymbol{E}^{(h)}(n,d))$$

所以线性方程组 $\boldsymbol{M}^{(h)}(n,d)x=\boldsymbol{E}^{(h)}(n,d)$在 $d<m$ 的条件下均无解，即结论成立。 $\square$

定理 4 若 $q^{(h)}(n)>q^{(h-1)}(N)$，则 $q^{(h)}(n)$是使得 $\mathrm{rank}(\boldsymbol{M}^{(h)}(n,q^{(h)}(n)))\neq\mathrm{rank}(\boldsymbol{M}^{(h-1)}(N,q^{(h)}(n)))$ 成立的最小整数，其中 $q^{(h-1)}(N)+1\leqslant n\leqslant N$。

证明

1) 首先，若 $q^{(h)}(n)>q^{(h-1)}(N)$，则由定理 2、推论 1 和定理 3，可以得到：

$$\mathrm{rank}(\boldsymbol{M}^{(h)}(n,q^{(h-1)}(N)))\neq\mathrm{rank}(\boldsymbol{M}^{(h)}(n,q^{(h-1)}(N))|\boldsymbol{E}^{(h)}(n,q^{(h-1)}(N)))$$

且

$$\mathrm{rank}(\boldsymbol{M}^{(h-1)}(N,q^{(h-1)}(N)))=\mathrm{rank}(\boldsymbol{M}^{(h)}(n,q^{(h-1)}(N)))$$

其中 $q^{(h-1)}(N)+1\leqslant n\leqslant N$。

即矩阵 $\boldsymbol{M}^{(h)}(n,q^{(h-1)}(N))$中行$(s_{h,u}\ s_{h,u+1}\ s_{h,u}s_{h,u+1}\ \cdots\ s_{h,n-1}\ s_{h,n-1}s_{h,n-2}\ \cdots\ s_{h,n-1}s_{h,u})$与前面各行线性相关；但矩阵 $\boldsymbol{M}^{(h)}(n,\ q^{(h-1)}(N)|\boldsymbol{E}^{(h)}(n,q^{(h-1)}(N))$ 中行$(s_{h,u}\ s_{h,u+1}\ s_{h,u}s_{h,u+1}\ \cdots\ s_{h,n-1}\ s_{h,n-1}s_{h,n-2}\ \cdots\ s_{h,n-1}s_{h,u}\ s_{h,n})$与前面各行是线性独立的，因而 $s_{h,n}$ 与列 $\boldsymbol{E}^{(h)}(n,q^{(h-1)}(N))$中前面的元素是线性独立的，其中 $u=n-q^{(h-1)}(N)$。

2) 然而，$q^{(h)}(n)$ 是使得式 $\mathrm{rank}(\boldsymbol{M}^{(h)}(n,q^{(h)}(n)))=\mathrm{rank}(\boldsymbol{M}^{(h)}(n,q^{(h)}(n)|\boldsymbol{E}^{(h)}(n,q^{(h)}(n)))$成立的最小整数，因而 $q^{(h)}(n)$是使得如下条件 A）或条件 B）成立的最小整数

A) 或者 $s_{h,n}$ 与列 $\boldsymbol{E}^{(h)}(n,q^{(h)}(n))$中前面的元素线性相关；

B) 或者 $(s_{h,u}\ s_{h,u+1}\ s_{h,u}s_{h,u+1}\ \cdots\ s_{h,n-1}\ s_{h,n-1}s_{h,n-2}\ \cdots\ s_{h,n-1}s_{h,u})$，其中 $u=n-q^{(h)}(N)$，与矩阵 $M^{(h)}(n,q^{(h)}(n))$中前面的行线性独立。

由于条件 A）与条件 1）的结果相矛盾，因而条件 B）成立，即 $q^{(h)}(n)$是使得 $\mathrm{rank}(\boldsymbol{M}^{(h)}(n,q^{(h)}(n)))\neq\mathrm{rank}(\boldsymbol{M}^{(h-1)}(N,q^{(h)}(n)))$成立的最小整数。 $\square$

推论 2 若 $q^{(h)}(n)>q^{(h)}(n-1)$，则 $q^{(h)}(n)$是使得 $\mathrm{rank}(\boldsymbol{M}^{(h)}(n,\ q^{(h)}(n)))\neq\mathrm{rank}(\boldsymbol{M}^{(h)}(n-1,q^{(h)}(n)))$成立的最小整数。

由于 t 维 N 周期多重序列 s 的联合二次复杂度的计算等价于搜索使得式(3)成立的最小 m 值，因而由定理 4 以及推论 2 知道在矩阵 $\boldsymbol{M}^{(t)}(N,m)$中：

或者所有的行都是线性独立的，即此时不存在 $\sum_{j=1}^{t}(N-q^{(j)}(N))$ 维向量$\lambda=(\lambda_1\cdots\lambda_{\sum_{j=1}^{t}(N-q^{(j)}(N))-1}$ 1) 使得$\lambda C_i(N,m)=0,\ 0\leqslant i\leqslant p(m,m)$，成立；

或者存在某行和前面的行线性相关，一旦这样的行存在，例如第 k 行与前面的行线性相关，则此时必然有

$$\mathrm{rank}(\boldsymbol{M}^{(h)}(n,m))=\mathrm{rank}(\boldsymbol{M}^{(h)}(n,m)|\boldsymbol{E}^{(h)}(n,m))$$

即存在有 k 维向量$\lambda=(\lambda_1\cdots\lambda_{k-1}\ 1)$使得$\lambda C_i(n,m)=0$，$0\leqslant i\leqslant p(m,m)+1$，且

$$\sum_{j=1}^{h-1}(N-q^{(j)}(N))+n-m=k \tag{6}$$

因而算法设计的主要任务就是研究相关向量的存在性以及验证式(6)的有效性。

4 算法

本节给出一个用于求解多序列联合二次复杂

度的递归算法。递归过程首先由矩阵 $\boldsymbol{M}^{(t)}(N,m)$的第一行最左端的元素开始一直到最右端，再到第二行，依此类推。

初始时假设$\boldsymbol{\Lambda}(1)=[1]$，$\boldsymbol{I}(1)=0$。

1) 假设已经计算得到了 $h-1$ 个 N 周期分量序列 $s_j(j=1,2,\cdots,h-1)$的联合二次复杂度，要计算 h 个 N 周期分量序列 $s_j(j=1,2,\cdots,h)$的联合二次复杂度，由定理 1 只需在矩阵 $\boldsymbol{M}^{(h-1)}(N,q^{(h-1)}(N))$中先添加一个新行$(1\ s_{h,1}\ s_{h,2}\ s_{h,1}s_{h,2}\ \cdots\ s_{h,q^{(h-1)}(N)}\ \cdots\ s_{h,1}s_{h,q^{(h-1)}(N)})$，记新得到的矩阵为 $\boldsymbol{M}^{(h)}(q^{(h-1)}(N)+1,q^{(h-1)}(N))$，则此新矩阵中共有 $w=\sum_{j=1}^{h-1}(N-q^{(j)}(N))+1$ 行。

根据定理 2，$h-1$ 个分量序列 s_j，$j=1,2,\cdots,h-1$，与 $s^{(h,q^{(h-1)}(N)+1)}$ 的联合二次复杂度 $q^{(h)}(q^{(h-1)}(N)+1)$ 等于 $q^{(h-1)}(N)$的充要条件为如下的条件①或②成立。

① $$\mathrm{rank}(\boldsymbol{M}^{(h)}(q^{(h-1)}(N)+1,q^{(h-1)}(N)))\neq\mathrm{rank}(\boldsymbol{M}^{(h-1)}(N,q^{(h-1)}(N))) \tag{7}$$

② 若

$$\mathrm{rank}(\boldsymbol{M}^{(h)}(q^{(h-1)}(N)+1,q^{(h-1)}(N)))=\mathrm{rank}(\boldsymbol{M}^{(h-1)}(N,q^{(h-1)}(N))) \tag{8}$$

则

$$\mathrm{rank}(\boldsymbol{M}^{(h)}(q^{(h-1)}(N)+1,q^{(h-1)}(N)))=\mathrm{rank}(\boldsymbol{M}^{(h)}(q^{(h-1)}(N)+1,q^{(h-1)}(N))|\boldsymbol{E}^{(h)}(q^{(h-1)}(N)+1,q^{(h-1)}(N))) \tag{9}$$

矩阵 $\boldsymbol{M}^{(h)}(q^{(h-1)}(N)+1,q^{(h-1)}(N))$的结构表明：

若矩阵 $\boldsymbol{M}^{(h)}(q^{(h-1)}(N)+1,q^{(h-1)}(N))$的最后一行与前面的行线性独立，则式(7)成立。即不存在 w 维的向量 $\lambda=(\lambda_1\ \cdots\ \lambda_{w-1}\ 1)$ 使得等式 $\lambda C_i(q^{(h-1)}(N)+1,q^{(h-1)}(N))=0$ 成立，其中 $0\leqslant i\leqslant p(q^{(h-1)}(q^{(h-1)}(N)+1),q^{(h-1)}(q^{(h-1)}(N)+1))$。也即存在

$$j<p(q^{(h-1)}(q^{(h-1)}(N)+1),q^{(h-1)}(q^{(h-1)}(N)+1))$$

使得

$$d_{w,j}=\lambda C_j(q^{(h-1)}(N)+1,q^{(h-1)}(N))=1$$

且$\lambda C_i(q^{(h-1)}(N)+1,q^{(h-1)}(N))=0$，$0\leqslant i<j$。

若存在一个 $u(0\leqslant u<w)$，使得 $\boldsymbol{I}(u)=j$，则由引理 3 在 w 行可以确定一个相关向量λ，使得$\lambda C_i(q^{(h-1)}(N)+1,q^{(h-1)}(N))=0$, $0\leqslant i\leqslant j$。

否则若不存在一个 $u(0\leqslant u<w)$，使得 $\boldsymbol{I}(u)=j$，即在第 j 列不存在一行与前面的行线性相关(可以使用引理 2 的方法进行确定)，则定义第 w 行最终的相关向量为λ，并将其存储在矩阵$\boldsymbol{\Lambda}$的第 w 行。同时将 j 存储在矩阵 $\boldsymbol{I}$ 的第 w 个位置。随后添加一个新行，并以向量$(0\ \lambda)$来考察最左端的一列。

否则若存在 w 维的向量$\lambda=(\lambda_1\cdots\lambda_{w-1}\ 1)$使得等式 $\lambda C_i(q^{(h-1)}(N)+1,q^{(h-1)}(N))=0$ 成立，其中 $0\leqslant I\leqslant p(q^{(h-1)}(q^{(h-1)}(N)+1),q^{(h-1)}(q^{(h-1)}(N)+1))$。则令 $I(w)=-1$。同时式(9)等价于

$$\lambda\boldsymbol{E}^{(h)}(q^{(h-1)}(N)+1,q^{(h-1)}(N))=\lambda C_{p(q^{(h-1)}(N)+1,1)}(q^{(h-1)}(N)+1,q^{(h-1)}(N))=0 \tag{10}$$

若式(10)不成立，取 $q^{(h)}(q^{(h-1)}(N)+2)$ 为 $q^{(h-1)}(N)+1$，即添加一个新的分块 $B_{q^{(h-1)}(N)+1}(q^{(h-1)}(N)+2,\ q^{(h-1)}(N)+1)$。同时新矩阵 $\boldsymbol{M}^{(h)}(q^{(h-1)}(N)+2,q^{(h-1)}(N))$的最后一行为$(1\ s_{h,1}\ s_{h,2}\ s_{h,1}s_{h,2}\ \cdots\ s_{h,q^{(h-1)}(N)+1}\ \cdots\ s_{h,1}s_{h,q^{(h-1)}(N)+1})$。在新矩阵中共有 $w=\sum_{j=1}^{h-1}(N-q^{(j)}(N))+1$ 行。由于式(10)不成立，$s_{h,q^{(h-1)}(N)+1}$ 与列中的前面元素线性独立，即 $\boldsymbol{I}(w)=q^{(h-1)}(N)+1$，同时式(7)成立。随后添加一个新行，并以向量$(0\ \boldsymbol{\Lambda}(w))$验证最左边一列。

若(10)成立，则对矩阵 $\boldsymbol{M}^{(h)}(q^{(h-1)}(N)+1,\ q^{(h)}(q^{(h-1)}(N))$添加一个新行，同时取 $q^{(h)}(q^{(h-1)}(N)+2)$为 $q^{(h)}(q^{(h-1)}(N)+1)$。

2) 假设已经处理了 $h-1$ 个 N 周期分量序列 $s_j(j=1,2,\cdots,h-1)$以及第 h 个分量序列 s_h 的最初 n 个元素，其解为 $q^{(h)}(n)$。

$h-1$ 个 N 周期分量序列 s_j，$j=1,2,\cdots,h-1$，与第 h 个分量序列 s_h 的最初 $n+1$ 个元素的联合二次复杂度的计算过程与 1)的分析类似，但是当式(10)不成立时，存在着一些差异。

即当添加一个新的分块时，取 $q^{(h)}(n+1)$ 为 $q^{(h)}(n+1)+1$，此时新的矩阵中，共有 $w=n+1-q^{(h)}(n+1)+\sum_{j=1}^{h-1}(N-q^{(j)}(N))$ 行，且新矩阵的最后一行为

$$(1\ \ s_{h,n+1-q^{(h)}(n+1)}\ \ s_{h,n+2-q^{(h)}(n+1)}\ \ s_{h,n+1-q^{(h)}(n+1)}\ s_{h,n+2-q^{(h)}(n+1)}\ \cdots\ s_{h,n}\ \ s_{h,n-1}s_{h,n}\ \cdots\ s_{h,n+1-q^{(h)}(n+1)}s_{h,n})。$$

接下来由推论 2，只需要判断矩阵 $\boldsymbol{M}^{(h)}(n+1,q^{(h)}(n+1))$的秩是否等于矩阵 $\boldsymbol{M}^{(h)}(n,q^{(h)}(n+1))$的秩。

若 $\boldsymbol{I}(w)\neq-1$，则式(7)成立，添加一个新行。由于式(10)不成立，$s_{h,n+1}$ 与列中前面的元素线性独立，

即 $\boldsymbol{I}(w+1)=p(q^{(h)}(n+1),1)$，同时式(7)成立。随后另外添加一个新行(即 $q^{(h)}(n+1)=q^{(h)}(n+2)=q^{(h)}(n)+1$，该结果也可以通过定理 2 而获得)，以向量$(0\ \boldsymbol{\Lambda}(w+1))$考察最左边的一列。

否则在新添加的分块中由第 $j=p(q^{(h)}(n+1),1)$列开始以向量$\boldsymbol{\Lambda}(w)$进行验证。

若对于 $j<p(q^{(h)}(n+1),q^{(h)}(n+1))$ 有 $d_{w,j}=\lambda C_j(n+1,q^{(h)}(n+1))=1$，则式(7)成立。同时令 $\boldsymbol{I}(w)=j$，$\boldsymbol{\Lambda}(w)=\lambda$。

否则，令 $\boldsymbol{I}(w)=-1$，同时添加一个新的分块，并取 $q^{(h)}(n+1)$为 $q^{(h)}(n+1)+1$。

引理 2 若存在不同的 $\boldsymbol{I}(1)=j(1),\boldsymbol{I}(2)=j(2),\cdots,\boldsymbol{I}(k)=j(k)$ 以及$\boldsymbol{\Lambda}(1),\boldsymbol{\Lambda}(2),\cdots,\boldsymbol{\Lambda}(k)$，则 $j(k)$长的第 k 个行向量$\boldsymbol{\Lambda}(k)$与前面的 $k-1$ 个向量不线性相关。同时最初的 k 行是线性独立的。

证明 由$\boldsymbol{\Lambda}(1),\boldsymbol{\Lambda}(2),\cdots,\boldsymbol{\Lambda}(k)$的定义以及假设 $j(1),j(2),\cdots,j(k)$彼此不相同，知道 $d_{i,j(i)}=1,\ 1\leqslant i\leqslant k$，处于矩阵 $\boldsymbol{D}=[d_{i,j}]$（称为差值矩阵）的不同行以及不同列。因而由最初的 k 行以及列 $j(1),j(2),\cdots,j(k)$所形成的子矩阵是非退化的。由于 $\boldsymbol{D}$ 的每一行由矩阵 $\boldsymbol{M}^{(h)}(N,m)$的最初 k 行的线性组合构成，矩阵 $\boldsymbol{M}^{(h)}(N,m)$中的相应的子矩阵必定也是非退化的。因而矩阵 $\boldsymbol{M}^{(h)}(N,m)$中的最初 k 行是线性独立的。 □

由$\boldsymbol{\Lambda}(w)$与$\boldsymbol{\Lambda}(u)$的定义，可以得到如下结果。

引理 3 给定$\boldsymbol{\Lambda}(w)$且令 $d_{w,j}=1$，同时在第 u 行存在一个最终的多项式$\boldsymbol{\Lambda}(u)$，则$\boldsymbol{\Lambda}(u)$满足条件 $d_{u,j}=1$，即 $\boldsymbol{I}(u)=j$，则$\boldsymbol{\Lambda}(w)=(\boldsymbol{\Lambda}(u)\ \boldsymbol{0}_{w-u})+\boldsymbol{\Lambda}(w)$ 满足条件 $d_{w,k}=0$，$0\leqslant k\leqslant j$，$1\leqslant u<w$。

在下面的算法中，函数 pos(j,$\boldsymbol{I}$)返回矩阵 $\boldsymbol{I}$ 中下标 j 的位置。若 j 不属于矩阵 $\boldsymbol{I}$，则 pos(j,$\boldsymbol{I}$)=−1。

算法 (联合二次复杂度 $q^{(t)}(N)$)：

Input:

n:=1; $q^{(1)}(1)$:=0; $jump$:=0; $\boldsymbol{\Lambda}(1)$:=[1]; $\boldsymbol{I}(1)$:=0; h:=1;

Output: $q^{(t)}(N)$.

1: while $h<t+1$;

if $h>1$, then $n:=q^{(h-1)}(N)+1$;

2: while $n<N-q^{(h-1)}(N)+1$; \\添加一个新行

if$n=q^{(h-1)}(N)+1$,then $q^{(h)}(n):=q^{(h-1)}(N)$; goto 3;

else

$q^{(h)}(n):=q^{(h)}(n-1)$;

while $jump>0$;

I($n-q^{(h)}(n)$):=$p(q_{\text{ini}}+jump,1)$;

$jump:=jump-1$;

$n:=n+1$;

if $n>N-q^{(h-1)}(N)$

$q^{(h)}(N):=q^{(h)}(n-1)$;

$h:=h+1$;

goto 1;

$q^{(h)}(n):=q^{(h)}(n-1)$;

end while;

3: $\lambda:=(0\ \boldsymbol{\Lambda}(n-1-q^{(h)}(n))$;

for (j=1;$j<p(q^{(h)}(n),q^{(h)}(n))+1$; j++)

Temp1:=$\lambda C_j(n,q^{(h)}(n))$;

If Temp1=1, then

if $j\in I$, then

r:=pos(j,I);

$\lambda:=(\boldsymbol{\Lambda}(r)\quad \boldsymbol{0}_{n-q^{(h)}(n)-r})+\lambda$;

else

$\boldsymbol{I}(n-q^{(h)}(n)):=j$;

$\boldsymbol{\Lambda}(n-q^{(h)}(n)):=\lambda$;

$n:=n+1$;

goto 2;

else

Temp2:=$\lambda C_{p(q^{(h)}(n)+1,1)}\ (n,q^{(h)}(n))$;

if Temp2=0, then

$\boldsymbol{I}(n-q^{(h)}(n)):=-1$;

$\boldsymbol{\Lambda}(n-q^{(h)}(n)):=\lambda$;

$n:=n+1$;

goto 2;

else \\ 添加一个新的分块;

if $n:=q^{(h-1)}(N)+1$, then

jump:=1; $n:=n+1$;

$q^{(h)}(n):=q^{(h)}(n-1)+1$;

$\boldsymbol{I}(n-q^{(h)}(n)):=p(q^{(h)}(n),1)$;

goto 2;

else

$q_{\text{ini}}:=q^{(h)}(n)$;

4: $q^{(h)}(n):=q^{(h)}(n)+1$;

If $\boldsymbol{I}(n-q^{(h)}(n))\neq-1$, then goto 5;

else

for ($j=p(q^{(h)}(n),1)$; $j<p(q^{(h)}(n),q^{(h)}(n))$; j++)

if $\boldsymbol{\Lambda}(n-q^{(h)}(n))C_j(n,q^{(h)}(n))=1$,

then $\boldsymbol{I}(n-q^{(h)}(n)):=j$; goto 5;

if $\boldsymbol{\Lambda}(n-q^{(h)}(n))\ C_{p(q^{(h)}(n),q^{(h)}(n))}\ (n,q^{(h)}(n))=0$,

then goto 4;

```
        else
            I(n−q^(h)(n)):=p(q^(h)(n), q^(h)(n)); goto 5;
5:      jump:=q^(h)(n)−q_ini;
        n:=n+1;
        goto 2;
    end while
    h:=h+1;
end while
```

5　算法复杂度分析

前面所给算法的主要任务就是要研究相关向量的存在性以及验证式(6)的有效性。算法复杂度的增加主要由 2 部分构成，一个是添加了新的一行，另外是添加了一个新的分块，例如若已经处理了 $h-1$ 个分量序列，对于第 h 个分量序列，我们需要处理的元素总数为 $N-q^{(h-1)}(N)$，而每新处理一个元素，或者添加一个新行，或者添加一个新的分块，

添加一个新行，即 $q^{(h)}(n)=q^{(h)}(n-1)$时，需要以矩阵$\boldsymbol{\Lambda}$中所储存的对应元素λ乘以新得到的矩阵各列，以验证相关性，所需运算量最多为此时矩阵中的元素个数

$$\sum_{h=1}^{t}(N-q^{(h)}(n))\left(1+\frac{q^{(h)}(n)(q^{(h)}(n)+1)}{2}\right)$$

添加一个新的分块，即 $q^{(h)}(n)=q^{(h)}(n-1)+1$ 时，至多需要以矩阵$\boldsymbol{\Lambda}$中所存储的对应元素λ乘以新得到的矩阵的最后一个分块中的各列，以验证相关性，所需运算量最多为此时矩阵最后一个分块中的元素个数

$$\sum_{h=1}^{t}(N-q^{(h)}(n))q^{(h)}(n)$$

因而 t 维 N 周期序列 $s_j(j=1,2,\cdots,t)$的联合二次复杂度的计算至多需要

$$\sum_{h=1}^{t}\sum_{n=q^{(h-1)}(N)}^{N}(N-q^{(h)}(n))\left(1+\frac{q^{(h)}(n)(q^{(h)}(n)+1)}{2}\right)$$

个加法/乘法，其中 $q^{(0)}(N)=0$，$q^{(1)}(1)=0$。

由于算法的计算复杂度是 $q^{(h)}(n)$的三次函数，而 $q^{(h)}(n)$通常是序列长度的线性量级，因而估计算法的计算复杂度至多为序列长度的三次函数。特别地，注意到

$$(N-q^{(h)}(n))\left(1+\frac{q^{(h)}(n)(q^{(h)}(n)+1)}{2}\right)\qquad(11)$$

在 $q^{(h)}(n)$相对于 N 较小时，例如 $N=2^l$，而 $q^{(h)}(n)$位于 l 附近时，式(11)为 N 的线性函数；在 $q^{(h)}(n)$比较接近 N 时，式(11)趋于 0。进而，得到如下结论。

当所给的 h 个分量序列的二次复杂度相对于 N 都较小时，算法的复杂度最多为序列长度 N 的线性函数；

而当所给的 h 个分量序列的二次复杂度都接近于 N 时，算法复杂度为所求第一个分量序列的二次复杂度的计算复杂度，因而至多为序列长度 N 的线性函数。

算法中需要存储 t 个分量序列，矩阵 $\boldsymbol{I}$ 以及矩阵$\boldsymbol{\Lambda}$。

6　算法举例

设 $s^{(1,7)}$=(0111001), $s^{(2,7)}$=(0100000), $s^{(3,7)}$=(1010101)，则矩阵 $\boldsymbol{E}^{(3)}(7,4)=[0,0,1,X_8,0,0,0,1,0,1]^{\mathrm{T}}$，其中 X_8 可以根据具体的关系式加以确定，而矩阵 $\boldsymbol{M}^{(3)}(7,4)$构造如下：

$$\boldsymbol{M}^{(3)}(7,4))=\begin{pmatrix}1&0&1&0&1&1&0&1&1&1&0\\1&1&1&1&1&1&1&0&0&0&0\\1&1&1&1&0&0&0&0&0&0&0\\1&1&0&0&0&0&0&1&0&0&1\\1&0&1&0&0&0&0&0&0&0&0\\1&1&0&0&0&0&0&0&0&0&0\\1&0&0&0&0&0&0&0&0&0&0\\1&1&0&0&1&0&1&0&0&0&0\\1&0&1&0&0&0&1&0&1&0\\1&1&0&0&1&0&1&0&0&0&0\end{pmatrix}$$

具体求解过程如表 1 所示。

表 1　　算法在每一步的输出

h	n	$q^{(3)}(n)$	λ	jump	$\boldsymbol{I}$
1	1	0	[1]	0	[0]
1	2	1	[1]	1	[0 -1]
1	3	1	[1 1]	0	[0 1]
1	4	1	[0 1 1]	0	[0 1 -1]
1	5	3	[1 1]	2	[0 1]
1	6	3	[0 1 1]	1	[0 1 4]
1	7	3	[0 0 1 1]	0	[0 1 4 2]
2	4	3	[1 1 1 0 1]	0	[0 1 4 2 6]
2	5	4	[1 1 1 0 1]	1	[0 1 4 2 6]
2	6	4	[0 0 0 1 0 1]	0	[0 1 4 2 6 7]
2	7	4	[0 0 1 1 1 0 1]	0	[0 1 4 2 6 7 3]
3	5	4	[1 0 0 0 1 1 0 1]	0	[0 1 4 2 6 7 3 5]
3	6	4	[0 0 0 1 1 1 0 0 1]	0	[0 1 4 2 6 7 3 5 9]
3	7	4	[0 0 0 0 0 0 0 1 0 1]	0	[0 1 4 2 6 7 3 5 9 −1]

7 结束语

本文给出了一个用于求解任意有限域上多重周期序列联合二次复杂度的算法。接下来将进一步研究多重周期序列的其它联合复杂度的求解。

参考文献：

[1] CHAN A H, GAMES R A. On the quadratic spans of the de Bruijn sequences[J]. IEEE Trans Inform Theory, 1990,34(4):822-829.

[2] CHAN H, GAMES R A, RUSHANAN J J. On quadratic m-sequences[A]. Proc Fast Software Encryption[C]. Berlin, Germany, 1994. 166-173.

[3] KHACHATRIAN L H. The lower bound of the quadratic span of de Bruijn sequences[J]. Design, Codes and Cryptography, 1993, 3:29-32.

[4] PENZHORN W T. Quadratic complexity of binary sequences[A]. Proceedings of the 1998 South African Symposium on Communications and Signal Processing[C]. 1998. 175-180.

[5] RIZOMILIOTIS P, KOLOKOTRONIS N, KALOUPTSIDIS N. On the quadratic span of binary sequences[A]. Proc IEEE Int Symp Information Theory[C]. Yokohama, Japan, 2003. 377.

[6] YOUSSEF A M, GONG G. On the quadratic span of binary sequences[EB/OL]. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.34.9451,2000.

[7] RIZOMILIOTIS P, KOLOKOTRONIS N, KALOUPTSIDIS N. On the quadratic span of binary sequences[J]. IEEE Trans Inform Theory, 2005, 51(5): 1840-1848.

[8] FENG G L, TZENG K K. A generalized of the berlekamp-massey algorithm for multisequence shift-register synthesis with applications to decoding cyclic codes[A]. IEEE Trans Inform Theory[C]. 1991. 1274-1287.

[9] FENG X T, DAI Z D. Expected value of the linear complexity of two-dimensional binary sequences[A]. Sequence and Its Application[C]. 2005. 113-128.

[10] DING C, XIAO G, SHAN W. The stability theory of stream ciphers[A]. Lecture Notes in Computer Science[C]. Springer, Berlin, 1991.

[11] XING C P, Multi-sequences with almost perfect linear complexity profile and function fields over finite fields[J]. J Complexity, 2000, 16: 661-675.

[12] MEIDL W, NIDERREITER H. The expected value of joint linear complexity of periodic multisequences[J]. Journal of Complexity, 2003, 19: 61-72.

[13] WANG L. Lattice Bases Reduction Algorithm in Function Fields and Multiselquence Linear Feedback Shift-Register Synthesis[D]. PhD Thesis, Graduate School, USTC, Beijing, 2003.

[14] MEIDL W. Discrete fourier transform, joint linear complexity and generalized joint linear complexity of multisequences, sequence and its application[A]. SETA 2004[C]. 2005.101-112.

[15] DAI Z D, IMAMURA K, YANG J H. Asymptotic behavior of normalized linear complexity of multi-sequences[A]. SETA 2004[C]. 2005. 129-142.

[16] NIEDERREITER H. The probabilistic theory of the joint linear complexity of multisequences[A]. SETA 2006[C]. 2006. 5-16.

[17] DAI Z D. Multi-continued fraction algorithms and their applications to sequences[A]. SETA 2006[C]. 2006. 17-33.

作者简介：

董丽华（1977-），女，辽宁盘锦人，博士，西安电子科技大学副教授，主要研究方向为序列密码的设计与安全分析。

胡予濮（1955-），男，河南濮阳人，博士，西安电子科技大学教授、博士生导师，主要研究方向为信息与网络安全。

曾勇（1978-），男，土家族，湖南常德人，博士，西安电子科技大学副教授，主要研究方向为网络安全、无线传感器网络及安全。