Moti Yung
Peng Liu
Dongdai Lin
Jiwu Jing
(Eds.)

Information Security and Cryptology

Fourth International Conference, Inscrypt 2008
Beijing, China, December 2008
Short Paper Proceedings

State Key Laboratory of Information Security
Chinese Association for Cryptologic Research

Inscrypt 2008

SCIENCE PRESS
Beijing

Moti Yung
Peng Liu
Dongdai Lin
Jiwu Jing
(Eds.)

Information Security and Cryptology

Fourth International Conference, Inscrypt 2008
Beijing, China, December 2008
Short Paper Proceedings

SCIENCE PRESS
Beijing

内 容 简 介

本书是2008年12月在北京召开的第四届中国密码学与信息安全国际会议(The 4th China International Conference on Information Security and Cryptology-Inscrypt 2008)的短文论文集。Inscrypt系列国际会议是由信息安全国家重点实验室发起，与中国密码学会联合举办的高水平国际会议，每年在中国举办一次，该会议论文集由 Springer 出版社出版。本书收录了这次会议的短文 12 篇。主要内容包括密码算法、数字签名与认证、安全协议、密码实现与应用等。

本书可供从事密码学、信息安全、通信与信息系统、计算机应用技术等专业的科技人员和高等院校师生参考。

图书在版编目(CIP)数据

信息安全与密码学国际会议论文集：2008：英文/林东岱等主编. Information Security and Cryptology: Fourth International Conference, Inscrypt 2008. —北京：科学出版社, 2009

ISBN 978-7-03-024615-8

Ⅰ.①信… ②Ⅰ… Ⅱ.林… Ⅲ.①信息系统-安全技术-国际学术会议-文集-英文 ②密码-理论-国际学术会议-文集-英文 Ⅳ.TP309-53 TN918.1-53

中国版本图书馆 CIP 数据核字(2009) 第 079767 号

责任编辑：鞠丽娜 / 责任校对：柏连海
责任印制：吕春珉 / 封面设计：三函设计

科学出版社 出版
北京东黄城根北街 16 号
邮政编码：100717
http://www.sciencep.com

双青印刷厂 印刷

科学出版社发行 各地新华书店经销
*
2009 年 6 月第 一 版 开本：B5(720×1000)
2009 年 6 月第一次印刷 印张：11 1/2
印数：1—1 500 字数：230 000

定价：50.00 元

(如有印装质量问题，我社负责调换〈双青〉)

Preface

The Fourth China International Conference on Information Security and Cryptology (Inscrypt 2008) was co-organized by the State Key Laboratory of Information Security and by the Chinese Association for Cryptologic Research. The conference was held in Beijing, China in mid-december, and was further sponsored by the Institute of Software, the Graduate University of the Chinese Academy of Sciences and the National Natural Science Foundations of China.

Given its four year success, Inscrypt is now a tradition. It is, in fact, a leading annual international event in the area of cryptography and information security taking place in China. We are pleased to report the continuous support of the entire community: authors, attendees, committee members, reviewers, sponsors and organizers. This state of affairs reflects the fact that the research areas covered by Inscrypt are important to modern computing, where increased security, trust, safety and reliability are required. This need makes sure that the relevant research community, world wide, continues producing important fundamental, experimental and applied work in the wide areas of cryptography and information security research. It is not a surprise that the scientific program of Inscrypt 2008 covered numerous fields of research within these general areas.

The International Program Committee of Inscrypt 2008 received a total of 183 submissions from 23 countries and regions, from which only 28 submissions were selected for presentation as regular papers which are published by Springer in the series of Lecture Notes in Computer Science, and 12 submissions were selected as short paper presentations which are published in this proceedings. All anonymous submissions were reviewed by experts in the relevant areas and based on their ranking, technical remarks and strict selection criteria the papers were chosen to the various tracks. The selection to both tracks was a highly competitive process. We also note that reviews of submissions by committee members were hidden from their authors throughout the entire review process. We further noted that due to the conference format, many good papers have not been accepted regrettably.

Inscrypt 2008 was made possible by the joint efforts of numerous people and organizations worldwide. We take this opportunity to thank the Program Committee members and the external experts they employed for their invaluable help in producing the conference program. We further thank the conference Organizing Committee, the various sponsors and the conference attendees. Last but not least, we express our great gratitude to all the authors who submitted papers to the conference, the invited speakers and the session Chairs.

Moti Yung, Peng Liu and Dongdai Lin

December 2008

Inscrypt 2008

4th China International Conference on Information Security and Cryptology

Beijing, China December 15–17, 2008

Sponsored and organized by

State Key Laboratory of Information Security
(Chinese Academy of Sciences)
and
Chinese Association for Cryptologic Research

General Chairs

Dengguo Feng	SKLOIS, Chinese Academy of Sciences, China

Program Co-chairs

Moti Yung	Google Inc. and Columbia University, USA
Peng Liu	Pennsylvania State University, USA
Dongdai Lin	SKLOIS, Institute of Software, Chinese Academy of Sciences, China

Program Committee

Vladimir S. Anashin	Moscow University, Russia
Vijay Atluri	Rutgers University, USA
Marina Blanton	University of Notre Dame, USA
Zhenfu Cao	Shanghai Jiaotong University, China
Claude Carlet	INRIA,Universite Paris 8, France
Jean-Sebastien Coron	University of Luxembourg, Luxembourg
Marc Dacier	Symantec Research Labs Europe
Cunsheng Ding	Hong Kong University of Science and Technology, China
Jintai Ding	University of Cincinnati, USA
Stefan Dziembowski	University of Rome "La Sapienza", Italy
Jean-Charles Faugere	INRIA, France
Guang Gong	University of Waterloo, Canada
Qijun Gu	Texas State University, USA
Martin Hell	University of Lund, Sweden

Xuxian Jiang	NC State University, USA
Jiwu Jing	Graduate University of CAS, China
Brian King	Indiana University-Purdue University, Indianapolis, USA
Miroslaw Kutylowski	Wroclaw University of Technology, Poland
Chi-Sung Lai	National Cheng Kung University, Taiwan
DongHoon Lee	Korea University, Korea
Albert Levi	Sabanci University, Turkey
Jianhua Li	Shanghai Jiaotong University, China
Jie Li	University of Tsukuba, Japan
Ninghui Li	Purdue University, USA
Yingjiu Li	Singapore Management University, Singapore
Benoit Libert	Universite Catholique de Louvain, Belgium
Javier Lopez	University of Malaga, Spain
Xiapu Luo	The Hong Kong Polytechnic University, China
Bodo Moeller	Google Inc., Zurich
Mridul Nandi	NIST, USA
Peng Ning	North Carolina State University, USA
Eiji Okamoto	Tsukuba University, Japan
Ludovic Perret	LIP6/INRIA Paris-Rocquencourt, France
Giuseppe Persiano	University of Salerno, Italy
Raphael C.-W. Phan	Loughborough University, UK
Bimal K. Roy	Indian Statistical Institute, India
Kouichi Sakurai	Kyushu University, Japan
Bhavani Thuraisingham	University of Texas at Dallas, USA
Carmela Troncoso	K.U. Leuven, Belgium
Shabsi Walfish	Google Inc. USA
Huaxiong Wang	Nanyang Technological University, Singapore
Xiaoyun Wang	Shandong University, China
Chuankun Wu	Institute of Software, CAS, China
Shouhuai Xu	University of Texas at San Antonio, USA
Meng Yu	Western Illinois University, USA
Erik Zenner	Technical University of Denmark
Yuliang Zheng	University of North Carolina at Charlotte, USA
Jianying Zhou	Institute for Infocomm Research, Singapore
Sencun Zhu	The Pennsylvania State University, USA

Organizing Committee Co-Chairs

Jiwu Jing	SKLOIS, Graduate University of Chinese Academy of Sciences, China
Zhijun Qiang	Chinese Association for Cryptologic Research, China

Organizing Committee

Chuankun Wu	SKLOIS, Institute of Software of CAS, China
Daren Zha	SKLOIS, Graduate University of CAS, China
Xiaoyang Wen	SKLOIS, Graduate University of CAS, China
Aihua Zhang	SKLOIS, Graduate University of CAS, China

WEB/Registration

Yicong Liu	SKLOIS, Graduate University of CAS, China
Jingjing Wu	SKLOIS, Graduate University of CAS, China

Table of Contents

Part I

Stream Cipher and Elliptic Curve Algorithm

Cryptanalysis of Generalized Self-shrinking Generator

Juntao Gao[1], Yupu Hu[1], Yongzhuang Wei[2]

1 School of Telecommunication and Engineering of Xidian University, Xi'an, 710071, China
2 Guilin University of Electronic Technology, Guilin, 541004, China
jtgao@mail.xidian.edu.cn

Abstract. Generalized self-shrinking generator consists of a linear feedback shift register and a vector G. It has been shown that the generalized self-shrinking sequences have partial ability to resist all cryptanalytic technique in previous works. This paper presents a related key cryptanalysis on the generalized self-shrinking generator. The results show that the attack complexity can be sharply reduced from $O(2^{n-1} \times n^3)$ to $O(k \times 2^{[\frac{n}{k+1}]+1})$ by using k inappropriate keys. Furthermore, if an attacker can obtain n inappropriate keys, the secret key can be identified with the complexity of $O(n^3)$. To resist the new attack, we design a key initialization process for the GSS stream cipher.

Keywords: Generalized self-shrinking generator, Cryptanalysis, Stream ciphers

1 Introduction

Pseudorandom sequences have wide applications in communications and cryptography. A stream cipher scheme based on a sequence generator should have simple configuration, good pseudorandom properties and strong security to resist various cryptanalytic techniques.

The shrinking-like generator, such as shrinking generator [8] and the self-shrinking generator [6], have been proposed for many years. These generator can produce pseudorandom sequences with high linear complexity. Until now, there are not effective attacks on the two generators. Generalized self-shrinking generator(GSS) [1] was proposed by Yupu Hu and Guozhen Xiao. The design idea of GSS comes from the shrinkng generator and self-shrinking generator. The GSS applies only one maximal length linear feedback shift register(LFSR) and a vector G to produce the keystream bit b_i as follows:

Definition 1. *Let $a = a_0 a_1 \cdots$ be a binary m-sequence with the least period $2^n - 1$. Let n-dimensional vector $G = (g_0, g_1, \cdots, g_{n-1}) \in GF(2)^n$. Sequence $v = v_0 v_1 \cdots$ such that*

$$v_k = g_0 a_k + g_1 a_{k-1} + \cdots + g_{n-1} a_{k-n+1}$$

Output v_k, if $a_k = 1$, or no output is produced. In this way, a sequence $b(G) = b_0 b_1 \cdots$ is generated. We call the sequence $b(G)$ a generalized self-shrinking sequence (GSS). The sequences family $B(a) = \{b(G) = b_0 b_1 b_2 \cdots, G \in GF(2)^n\}$ is called the family of generalized self-shrinking sequences based on m-sequence a.

The generalized self-shrinking sequence $b(G) = b_0 b_1 b_2 \cdots$ has lots of good pseudorandom property, such as,

- $b(G) = 000\cdots$ if and only if G =(0,0,$\cdots$,0);
- $b(G) = 111\cdots$ if and only if G =(1,0,$\cdots$,0);
- Except for $G = (0, 0, \cdots, 0)$ and $G = (1, 0, \cdots, 0)$,$b(G)$ is balanced in each consecutive 2^{n-1} bits;
- The family $B(a)$ consists of a n-dimensional linear space, and $|B(a)| = 2^n$.
- The least period of $b(G)$ divides 2^{n-1}, no more than $\frac{1}{4}$ of the sequences from $B(a)$ have the least periods less than 2^{n-1}.
- For each $b(G) \in B(a)$, where $G \neq (0, 0, \cdots, 0), G \neq (1, 0, \cdots, 0)$, the run length of sequence $b(G)$ is less than $n^2 - 3n + 3$.

If the least period of a GSS in **Definition 1** equals 2^{n-1}, then the GSS is called as a GSS with the maximal least period. In [2], the authors presented a simple method for producing GSS with the maximal least period. Obviously the linear complexity of the GSS with the maximal least period is larger than 2^{n-2}. The authors denoted that the GSS with maximal least period could be used for stream cipher with three secret keys: the initial state of LFSR, the coefficients of feedback polynomial of LFSR and the vector G. In [3], [4], the authors proposed a cryptanalysis on the stream cipher based on GSS respectively. In [3], the authors discussed the security of GSS from the following two aspects:

- If both the vector G and the feedback polynomial of LFSR are known, then the complexity of attack is $O(2^{0.694n})$ under an improved clock-guessing attack, where n is the length of the LFSR in GSS.
- If the feedback polynomial of LFSR is known, then for a stream cipher based on GSS with 61-stage LFSR, the complexity is $O(2^{56})$ under a fast correlation attack.

In [4], the authors discussed the security of GSS from the following three aspects:

- If both the vector G and the feedback polynomial of LFSR are known, the complexity of attack is $O(2^{n-l} \times l^3)$, where $l \leq \frac{n}{2}$ and n is the length of the LFSR in Generator.
- If the feedback polynomial of LFSR is known, the complexity of attack is $O(2^{2n-l} \times l^3)$, where $l \leq n$ and n is the length of the LFSR in Generator.
- If the vector G is known, the complexity of attack is $O(\phi(2^n-1) \times n^2 \times 2^{2n-l})$, where $l \leq n$, $\phi(\bullet)$ denotes the Euler functions, and n is the length of the LFSR in Generator.

Notation: On condition that the vector G and the feedback polynomial of LFSR are known, the cryptanalytic techniques in[3] and [4] are significantly different, which leads to the complexity is different accordingly. The more details could be found in [3] and [4].

The above results show that, even if the vector G is known, the stream cipher based on GSS is strong enough to withstand all given cryptanalytic techniques. For example, in[3], when n=128, the complexity of attack achieves $O(2^{88.83})$ with the vector G known. This paper presents a much faster attack on the stream cipher based on GSS. More specifically, our results can be shown as follows:

If both the vector G and the feedback polynomial of LFSR are known, then the complexity of attack can be significantly reduced from $O(2^{0.694n})$ to $O(k \times 2^{[\frac{n}{k+1}]+1})$ by using a few other keystreams, where $k = 5$ for $60 \leq n \leq 128$.

In Section 2, we give several Lemmas on the GSS. It is demonstrated that, for the GSS generator with two secret keys: the initial state of LFSR and the vector G, the complexity of exhaust attack is $O(\frac{\phi(2^n-1)}{n} \times 2^{n-1} \times n^3)$ other than $O(\frac{\phi(2^n-1)}{n} \times 2^{2n})$. In Section 3, the new attack on GSS with vector G known is given. We argue that the attacker can obtain the initial state of LFSR by using a few different keystreams which are produced by different G . Furthermore, if the attacker can obtain enough keystreams, the complexity of attack can be significantly decreased to $O(n^3)$. In Section 4, we design a key initialization process to resist the new attack. The key initialization process is simple and effective to produce the running keys.

2 Several Lemmas on GSS

Definition 2. *Let $a = a_0a_1a_2\cdots$ is an m-sequence over GF(2) with the least period $2^n - 1$. Let T=$\{t(u)|u = 0, 1, 2, \cdots\}$ be a set such that $0 \leq t(0) \leq t(1) \leq \cdots$, $a_{t(u)} = 1$ for each $t(u) \in T$, and $a_t = 0$ for each $t \notin T$. We call $t(u)$ the u-th output time of family $B(a)$.*

Lemma 1. *Suppose $(a_0, a_1, \ldots, a_{n-1})$ is the initial state of LFSR in Definition1 and $a_0 \neq 0$, then there must exist an equivalent state $(a_{t(0)}, a_{t(0)+1}, \ldots, a_{t(0)+n-1})$, which can produce same GSS with identical vector G.*

The Lemma 1 is obvious by the operation rules of GSS generator. Therefore, we always assume that the first bit of initial state of GSS generator is 1.

Lemma 2. *With the feedback polynomial and the initial state of LFSR known, the attacker can obtain the vector G with a complexity of $O(n^3)$.*

Proof. With the feedback polynomial and the initial state of LFSR known, the attacker can obtain all information on m-sequence a. According to the operation rules of GSS, the attacker can list lots of linear equations on the vector G by the given keystream bits $b_0, b_1, \cdots, b_j$, that is,

$$\begin{cases} g_0 a_{t(0)} + g_1 a_{t(0)-1} + \cdots + g_{n-1} a_{t(0)-n+1} = b_0 \\ g_0 a_{t(1)} + g_1 a_{t(1)-1} + \cdots + g_{n-1} a_{t(1)-n+1} = b_1 \\ \cdots \\ g_0 a_{t(j)} + g_1 a_{t(j)-1} + \cdots + g_{n-1} a_{t(j)-n+1} = b_j \\ \cdots \end{cases}$$

By the m-sequence theory, it is not difficult to find n linearly independent equations, which can be solved by Gaussian elimination method with complexity $O(n^3)$. □

Lemma 3. *For the GSS with three key, i.e., the vector G, the feedback polynomial and initial state of LFSR, the complexity of exhaustive attack achieves $O(\frac{\phi(2^n-1)}{n} \times 2^{n-1} \times n^3)$.*

Proof. By Lemma 1, the attacker knows a_0=1, therefore, the complexity of exhaustive search about the remaining bits of the initial state is $O(2^{n-1})$. By Lemma 2 and the number of n-stage primitive polynomials, the complexity of exhaustion attack is $O(\frac{\phi(2^n-1)}{n} \times 2^{n-1} \times n^3)$. □

By Lemma 3, the security of GSS nearly depends on the feedback polynomial and the initial state of LFSR. The vector G seems not important compared with the feedback polynomial and the initial state. This paper shows that the vector G is also important. If the attacker can obtain a few specific G's then the GSS will become vulnerable. Therefore, the vector G should be chosen carefully.

3 Our Attack on GSS

Here, we assume the users choose the feedback polynomial $f(x)$ and the initial state of LFSR as the keys of GSS stream cipher, however the vector G is public. On the other hand, the attacker can obtain other several keystreams which are encrypted by using same $f(x)$ and $(a_{t(0)}, a_{t(0)-1}, \ldots, a_{t(0)-n+1})$ but different vectors G. Our assumptions is reasonable, because different plaintexts can not be encrypted by same keystreams, otherwise the ciphertexts can be broken under the known plaintext attack.

Suppose the attacker has obtained several former bits of j different keysreams, such as

$$\begin{aligned} b(G^{(1)}) &= b_0^{(1)}, b_1^{(1)}, \cdots, b_k^{(1)} \\ b(G^{(2)}) &= b_0^{(2)}, b_1^{(2)}, \cdots, b_k^{(2)} \\ &\cdots \\ b(G^{(j)}) &= b_0^{(j)}, b_1^{(j)}, \cdots, b_k^{(j)} \end{aligned}$$

Obviously, these keystreams are encrypted by same $f(x)$ and $(a_0, a_1, \cdots, a_{n-1})$ but different vectors G respectively.

Theorem 1. *Let $M=(G^{(1)}, G^{(2)}, \cdots, G^{(j)})^{\mathrm{T}}$, where $G^{(i)} = (g_0^{(i)}, g_1^{(i)}, \cdots, g_{n-1}^{(i)})$, $i = 0, 1, \cdots, j$. If rank(M)=n, then the attacker can obtain the feedback polynomial $f(x)$ and the initial state $(a_{t(0)}, a_{t(0)-1}, \ldots, a_{t(0)-n+1})$ of LFSR with the complexity $O(n^3)$.*

Proof. By the Definition 1, we have following equations:

$$\begin{cases} g_0^{(1)}a_{t(0)} + g_1^{(1)}a_{t(0)-1} + \cdots + g_{n-1}^{(1)}a_{t(0)-n+1} = b_0^{(1)} \\ g_0^{(2)}a_{t(0)} + g_1^{(2)}a_{t(0)-1} + \cdots + g_{n-1}^{(2)}a_{t(0)-n+1} = b_0^{(2)} \\ \cdots \\ g_0^{(j)}a_{t(0)} + g_1^{(j)}a_{t(0)-1} + \cdots + g_{n-1}^{(j)}a_{t(0)-n+1} = b_0^{(j)} \end{cases} \quad (1)$$

Since rank(M)=n, the unique solution to equation (1) can be obtained with the complexity $O(n^3)$.

The second procedure is to obtain the feedback polynomial $f(x)$, we require $2n$ consecutive bits in the m sequence a. By the given initial state and the previous keystreams, we can list at most two equation groups as (1) to find the unknown n bits $(a_{t(0)+1}, a_{t(0)+2}, \cdots, a_{t(0)+n})$ by the Gaussian eliminate methods. By Berlekamp-Massey algorithm [7], the feedback polynomial $f(x)$ can be found. Since the complexity of Berlekamp-Massey algorithm is $O(n^2)$, the total complexity of the second procedure is $O(n^3)$. □

Although the above method can obtain the secret keys of GSS, it requires at least n linearly independent vector $G^{(i)}$. This gives a hint that, for a secret key, the number of $G^{(i)}$ used in the encryption process should be less than n. However, in the following, it is demonstrated that ,even if less amount of $G^{(i)}$ are used, the attacker can still launch an attack on the GSS. Exactly, for $60 \leq n \leq 128$, we only need 4 keystreams generated by 4 specific $G^{(i)}$ respectively to obtain the initial state of LFSR in GSS.

Theorem 2. *For a GSS with the vector G public, the attacker can obtain the $(a_{t(0)}, a_{t(0)-1}, \cdots, a_{t(0)-n+1})$ with the complexity of $O(k \times 2^{[\frac{n}{k+1}]+1})$ by using k inappropriate keystreams generated by k vectors G and same initial state.*

To prove the Theorem 2, we require following several Lemmas:

Lemma 4. *Let $b(G)$ is a GSS sequences generated by initial state $(a_{t(0)}, a_{t(0)-1}, \ldots, a_{t(0)-n+1})$ and vector $G = (g_0, g_1, \cdots, g_{n-1})$. Also,$b(G')$ is generated by initial state $(a_{t(0)}, a_{t(0)-1}, \ldots, a_{t(0)-n+1})$ and vector $G' = (g'_0, g'_1, \cdots, g'_{n-1})$, where $g_j \oplus g'_j$=0 for $j = 0, 1, \cdots, i-1, i+1, \cdots, n-1$, and $g_i \oplus g'_i$=1. then $b_k \oplus b'_k = a_{t(k)-i}$.*

The Lemma 4 is obvious from the Definition1.

Lemma 5. *Let $a = a_0, a_1, \cdots, a_n, \cdots$, be an m sequence with the least period $2^n - 1$. the number of n-dimensional vector $(a_{t(i)}, a_{t(i)-1}, \cdots, a_{t(i)-n+1})$ with the run-length of '0' is not greater than k is denoted as $n(k)$. Then*

- *If n mod $(k+1)$=0, then $n(k) \geq 2^{\frac{nk}{k+1}}(2^{-2}+1)^{\frac{n-k-1}{k+1}}$*
- *If n mod $(k+1) = j, j > 0$,$n(k) \geq 2^{[\frac{n}{k+1}]\times k+j-1}(2^{-2}+1)^{[\frac{n}{k+1}]-1}$*

Proof. In an m sequences with the least period 2^n-1 , all n-dimensional state but $(0, 0, \cdots, 0)$ are included, however, we only consider the number of n-dimensional

state V, whose first entries equal to '1' and the run-length of '0' is not greater than k. All of the n-dimensional state V consists of a set, which is denoted by $V(k)$.

(1) $n \bmod (k+1) = 0$

We consider a specific n-dimensional state $V = (10\cdots010\cdots0\cdots10\cdots0)$. The state V consists of $\frac{n}{k+1}$ blocks, and each block consists of '$10\cdots0$'. Obviously the state $V \in V(k)$. In the following, we consider the state which is obtained by changing some bits in the state V.

(i) There are $\frac{k}{k+1}n$ 0's in the state V, and the run-length of 0's is not greater than k. Now, Arbitrarily choose i 0's and change any one of them to '1', then we will obtain $2^{\frac{nk}{k+1}}$ n-dimensional states, all of which are belong to $V(k)$.

(ii) There are $\frac{n}{k+1}$ 1's in the state V. Choose any one '1' but the $a_{t(i)}$, and change any one of them to '0', on the other hand, change the '0' in the two sides of the original '1' to 1, for example, change '010' to '101'. By the above transformation, we can obtain many of n-dimensional states, all of which are belong to $V(k)$. the sum of the number of the states obtained is as follows:

$$\binom{\frac{n}{k+1}-1}{1}2^{\frac{kn}{k+1}-2}+\binom{\frac{n}{k+1}-1}{2}2^{\frac{kn}{k+1}-2\times 2}+\cdots+\binom{\frac{n}{k+1}-1}{\frac{n}{k+1}-1}2^{\frac{kn}{k+1}-2\times(\frac{n}{k+1}-1)} \tag{2}$$

The value of (2) equals to

$$2^{\frac{nk}{k+1}}(2^{-2}+1)^{\frac{n}{k+1}-1}-2^{\frac{kn}{k+1}}$$

The states set obtained by methods (i) and the state set obtained by methods (ii) are not overlapped. the total number of states is

$$2^{\frac{nk}{k+1}}(2^{-2}+1)^{\frac{n}{k+1}-1}$$

(2) $n \bmod (k+1) = j, j > 0$

we still consider a specific n-dimensional states $V = (10\cdots0\cdots10\cdots0\underline{1*\cdots*})$, where '*' represents '0' or '1', that is the state V consists of $[\frac{n}{k+1}]$ bit strings '$10\cdots0$' and the bit string $\underline{1*\cdots*}$ with a length of j.

(i) Arbitrarily choose i '0' in the former $n-j$ bits of the state V, and transform them to '1'. Then the number of vector satisfy the condition is

$$2^{[\frac{n}{k+1}]\times k+j-1}$$

(ii) Arbitrarily choose i '1' but the $a_{t(i)}$ in the former $n-j$ bits of the state V, and change any one of them to '0',on the other hand, change the '0' in the two sides of the '1' chosen to '1', for example, change '010' to '101'. By the above transformation, we can obtain many of n-dimensional states, all of which are belong to $V(k)$. The total number of n-dimensional states obtain is

$$2^{[\frac{n}{k+1}]\times k+j-1}(2^{-2}+1)^{[\frac{n}{k+1}]-1}-2^{[\frac{n}{k+1}]\times k+j-1}$$

So, if $n \bmod (k+1) = j, j > 0$, then there are at least$2^{[\frac{n}{k+1}]\times k+j-1}(2^{-2}+1)^{[\frac{n}{k+1}]-1}$ n-dimensional states in $V(k)$. □