分类号 TN918, N93 密级 公开

U D C

西安交通大學

博 士 学 位 论 文

Analyses and New Designs of Digital Chaotic Ciphers

数字化混沌密码的分析与设计

Shujun Li/李树钧

指导教师姓名 蔡元龙 教 授 西安交通大学

牟轩沁 教 授 西安交通大学

申请学位级别 博 士 专业名称 信息与通信工程

论文提交日期 2003年4月 论文答辩日期 2003年6月

答辩委员会主席 Yumin Wang/王育民(西安电子科技大学)

评 阅 人 Guanrong Chen/陈关荣(香港城市大学)

Kwok-Wo Wong/黄国和(香港城市大学)

Yumin Wang/王育民(西安电子科技大学)

Jianxue Xu/徐 健 学(西 安 交 通 大 学)

Qinye Yin/殷 勤 业(西 安 交 通 大 学)

2003年4月 (最后修订：2009年5月23日)

Title: Analyses and New Designs of Digital Chaotic Ciphers
Major: Information and Communication Engineering
Specialty: Chaotic Cryptography
Name: Shujun LI Signature: _____
Supervisor: Prof. Yuanlong CAI Signature: _____
Prof. Xuanqin MOU Signature: _____
Type: Applied Foundation

Abstract (英文摘要)

Since 1980s, the idea of using digital chaotic systems to design new ciphers has attracted more and more attention. The use of chaos in cryptography depends on the natural relation between the two research areas: strong dynamical properties of chaotic systems implies strong cryptographical properties of cryptosystems, and the basic way to make cryptosystems have good strong cryptographical properties implies quasi-chaos (we even can find the phantom of chaos in Shanon's classic paper on theory of secrecy systems). Basically speaking, there are two paradigms of applying chaos for secure applications: analog chaotic secure communications (mainly based on chaos synchronization technique) and digital chaotic ciphers realized in computers. This dissertation only focuses on digital chaotic ciphers, i.e., the area lying between chaos theory and pure cryptography in finite-state (digital) world.

The first boom of research on digital chaotic ciphers occurred near 1990, but decayed rapidly after some negative works on the security of proposed ciphers. The research boom returned in late 1990s, and much more contributions are made to get many promising achievements. Although a number of recently-proposed digital chaotic ciphers have been cryptanalyzed, many others have not been effectively attacked. Also, some general ways to digital chaotic ciphers have been proposed and more careful considerations are made to ensure their security, such as the chaotic block ciphers based on fixed or dynamic S-boxes.

However, as an important issue in the design of digital chaotic ciphers, dynamical degradation of digital chaotic systems has not been seriously considered by most designers of digital chaotic ciphers. One task of this dissertation is to emphasize significance of this issue and to give some initial investigations.

Our contributions in this dissertation involve the following three aspects about digital chaotic ciphers: theoretical analyses of dynamical degradation of digital chaotic systems, cryptanalyses of digital chaotic ciphers, and new proposals of digital chaotic ciphers. The main achievements contained in this disserta-

tion are as follows:

1. Because of piecewise linear chaotic maps (PWLCM-s) have perfect dynamical properties and can be realized simply in both hardware and software, they are widely used in digital chaotic ciphers. Aiming at digital PWLCM-s, a series of measurable dynamical indicators are proposed to quantitatively measure the dynamical degradation of PWLCM-s in (fixed-point) finite precision. Rigorous theoretical analyses are given to show how to calculate the dynamical indicators. The proposed dynamical indicators are used to compare performances of different remedies to dynamical degradation of digital chaotic systems, and are used to find security defects in digital chaotic ciphers.

2. Based on the theoretical results on dynamical degradation of PWLCM-s (see above), some digital chaotic stream ciphers proposed by Hong Zhou et al. are cryptanalyzed with weak-key analyses. Possible solutions to enhance concerned ciphers are compared and some ones are suggested as practical remedies.

3. In 1999, E. Alvarez et al. proposed a chaotic cipher based on searching plaintext in a pseudo-random sequence generated from chaotic systems, but soon it was broken by G. Alvarez et al. in 2000. This dissertation proposes a modified scheme to enhance its security by avoiding some essential defects in original cipher.

4. In 1998, M. S. Baptista proposed a searching based chaotic cipher, which attracted much attention after its proposal. Some cryptanalytic works and modifications are made in recent years. This dissertation points out the deficiency of an attack proposed by Goce Jakimoski and Ljupčo Kocarev, and presents a remedy to resist Jakimoski-Kocarev attack and some new attacks proposed by G. Alvarez in 2003.

5. This dissertation analyzes problems of a probabilistic chaotic cipher proposed by S. Papadimitriou et al. in 2001 and finds it is insecure and impractical. Also, some wrong results in this cipher are pointed out and rectified.

6. J.-C. Yen and J.-I. Guo et al. proposed several chaotic image encryption methods in recent years. This dissertation breaks two Yen-Guo chaotic image encryption methods with known/chosen plaintext attack, and finds more security problems in one method.

7. Based on theoretical results on digital chaotic systems and cryptanalyses of several recently-proposed chaotic ciphers, this dissertation proposes a new

chaotic PRBG and uses it to design chaotic stream ciphers with better overall performances. The proposed chaotic PRBG can be used instead of LFSR in conventional stream-cipher cryptography to construct more flexible ciphers. This dissertation also proposes a fast chaotic cipher employing multiple (2^n) chaotic systems. This fast cipher is specially designed to fulfill needs of real-time video encryption.

Keywords: chaotic system; cryptography; cryptanalysis; stream cipher; block cipher; pseudo-random sequence; piecewise linear chaotic map (PWLCM); image encryption

论文题目: 数字化混沌密码的分析与设计
专　　业: 信息与通信工程
研究方向: 混沌加密
博 士 生: Shujun Li/李树钧　　(签名) _____
指导教师: 蔡元龙 教授　　(签名) _____
　　牟轩沁 教授　　(签名) _____
论文类型: 应用基础

中文摘要

自从二十世纪80年代以来，应用数字化混沌系统构造新型密码系统的想法收到了越来越多的关注。这样一个想法来自混沌理论和纯粹密码学之间的自然联系：强混沌系统的动力学特性大致对应着高强度密码系统的某些安全特征，而具有良好混合性的传统密码系统又暗示着拟混沌现象（我们甚至可以在Shanon关于保密系统理论的经典文章里面看到混沌的影子）。一般而言，目前存在两种不同的混沌密码范式，一种是模拟化的混沌保密通信系统（大多基于混沌同步技术），另外一种是在数字计算机上实现的混沌密码系统。本文主要讨论数字化混沌密码系统，也就是位于混沌理论和纯粹密码学之间的研究领域。

早在1990年前后，数字化混沌密码就曾经有过短暂的繁荣时期，但是很快就由于负面的密码分析工作而很快消退。第二次繁荣在2000年前后兴起，有大量的研究结果发表，出现了不少很有希望的进展。尽管相当数量新提出的数字混沌密码还是很快被分析了，但是仍有很多直到目前尚未遇到真正有效的攻击。另外，一些设计混沌密码的通用方法已经出现，这些方法往往对系统的安全性做了相当细致的分析，从而具有较高的实用价值和安全性，这样的例子如使用由混沌系统生成的(固定的或者动态更新的)S盒的分组密码。

但是，作为数字化混沌密码设计中的一个重要问题，数字化混沌系统的动力特性退化却没有受到大多数混沌密码设计者应有的重视。本文的目的之一就是要强调这个问题的重要性，并给出一些初步的分析结果（参考本文成果的第一点）。

本文关于数字化混沌密码的研究涉及以下三个方面：数字化混沌系统动力特性退化的理论分析；数字化混沌密码的分析；和新的混沌密码的设计方法。具体说来，本文的主要成果包括如下几个方面：

1. 由于逐段线性混沌映射(PWLCM)具有优良的动力学特性，并且具有简单的软硬件实现方法，因此在数字化混沌密码的设计中得到了广泛的应用。针对数字化逐段线性混沌映射，本文提出了一组动力学指标，可以用来定量地衡量数字混沌系统动力特性退化的程度。严格的理论分析给出了准确计算这些

动力学指标的办法。这些指标可以用来对几种改善动力特性退化的方法做比较评价，也可以用来发现数字化混沌密码系统中的安全缺陷。

2. 基于上面研究的逐段线性混沌映射动力学特性退化的理论结果，本文对复旦大学周红等人提出的几种混沌流密码进行了弱密码分析，并在此基础上提出了一类密码分析方案。本文对一些可能的改进方案进行了比较，给出了几种可行的改进方案。

3. E. Alvarez等人于1999年提出了一种混沌密码，该密码通过在一个伪随机序列中搜索明文的办法来生成密文。但是该密码很快就被G. Alvarez等人在2000年破解。本文分析了原密码方案不安全的几个本质缺陷，提出了一种改进方案以抵抗已知攻击。

4. M. S. Baptista等人于1998年提出了另外一种基于搜索的混沌密码，该密码方案在提出后受到了广泛关注，近年来陆续有密码分析和改进方案发表。本文分析了Goce Jakimoski与Ljupčo Kocarev提出的一类攻击，指出它的性能并不够强，并提出了一种措施改进原密码方案使之可以抵抗Jakimoski-Kocarev攻击和G. Alvarez于2003年新提出的攻击。

5. 针对S. Papadimitriou等人于2001年提出的一种概率分组密码，我们的分析指出该密码既不实用也不安全。我们同时更正了该密码中存在一些错误分析。

6. J.-C. Yen和J.-I. Guo等人近年来陆续提出了几种混沌图像加密算法。本文对其中的两种算法进行了密码分析，发现它们在已知/选择明文攻击下是不安全的。

7. 在前面关于数字化混沌系统的理论分析和对部分混沌密码的密码分析的基础上，本文提出了一类新的混沌伪随机比特发生器，并将其代替LFSR来构造混沌流密码，分析表明CCS-PRBG有希望用来设计性能优于其他混沌流密码和基于LFSR的流密码的密码系统。另外，本文还提出了一种基于多(2^n个)混沌系统的快速密码系统，该系统针对实时视频加密的需求做了专门设计，可以满足视频加密在速度和格式上的需求。

关键词： 混沌系统；密码学；密码分析；流密码；分组密码；伪随机序列；逐段线性混沌映射(PWLCM)；图像加密

Table of Contents (目录)

Part I

Digital Chaotic Ciphers: State-of-the-Art and Theoretical Issues

Chapter 1
Introduction

§1.1 Research Background and Significance

As a surprising branch in natural science, chaos theory is developed since 1960s (and established in 1970s) with efforts from many different research areas*, such as mathematics[2–5], physics[6–8], biology[9], chemistry[10, 11] and engineering[12, 13], etc.[14, 15]. The most well-known characteristic of chaos is so-called "butterfly-effect" (formally, the sensitivity to the initial conditions and/or control parameters, or positivity of Lyapunov exponent)[12, 13, 16], which makes the chaotic orbits generated by deterministic equations become entirely "unpredictable" as time elapses.

Some researchers have pointed out that there exists tight relationship between chaos and cryptography[17–22]. Many fundamental characteristics of chaos, such as the ergodicity, mixing and exactness property† and the sensitivity to initial conditions, can be connected with the "confusion" and "diffusion" property in cryptography. So it is a natural idea to use chaos to enrich the design of new ciphers. In addition, since many chaotic systems have been extensively studied in past years, there are plenty of theoretical results can be used to make performance analyses on the designed chaotic ciphers. From 1989, many chaotic ciphers have been proposed and analyzed[18–22, 24–141].

Interestingly, the idea of using chaos in cryptography can be traced back to Shanon's classic paper titled "Communication Theory of Secrecy Systems" published in 1949[142]. Of course, he could not use the unborn word "chaos"; he just mentioned that the **good mixing transformations** used in a good secrecy systems can be constructed by the basic **rolled-out and folded-over** operations, where **good mixing transformations** can be considered as **chaotic maps** bounded in limited phase space with positive Lyapunov exponent (consider the **stretch-and-fold** mechanism in chaotic systems, such as Baker map and Smale horseshoe)[14, 15]. In [19], Ljupčo Kocarev et al. demonstrated how to construct a DES-like block cipher using chaotic maps in a general way. In very recent years, the idea of using chaos to generate S-boxes and then to design new ciphers has been investigated by Ljupčo Kocarev et al. in [105, 108], Jesús Urías in [55] and us in [112]. The above works have shown that chaos can be used to design ciphers with a similar

*Actually, pioneering research on chaos can be retrospected to H. Poincaré's work in 1890s, when he found the complexity of three-body celestial system[1].

†For definitions of related concepts, please refer to [23].

way to most basic techniques used by traditional cryptographers for many years.

On the other side, any good cipher can be regraded as a chaotic or pseudo-chaotic system from an algorithmic point of view[143], since perfect cryptographic properties are ensured by pseudo-random disorder generated from deterministic encryption operations (such as mod p operation and the nonlinear S-boxes/P-boxes in block ciphers[144, 145]), which is just like chaos generated from complex dynamical systems[17]. In [21], Marco Götz et al. have shown some conventional stream ciphers can exhibit chaotic behaviors. In fact, many chaotic systems employ mod p function when they are realized in digital computers with finite computing precision[86, 89, 96, 119, 120, 124]. As we know, mod p operation is a common component in most digital ciphers.

From the above discussion, I believe that the research on chaotic cryptography will be helpful to benefit the conventional cryptology and open a broader road to the design of good ciphers. In the following chapters, we will find that research on chaotic ciphers can also benefit chaos theory in discretized time and discretized space (i.e., in the digital world).

Apparently, chaotic cryptography is a multidisciplinary filed covering many different areas: nonlinear dynamics, cryptology, communications and etc. As a result, except a small number results are published in cryptology-related conferences and journals[20, 22, 28, 47, 57–59, 62–64, 66, 67, 109], most papers are published out of the security community (especially in physics and electronic engineering). See the references list of this dissertation, we can easily find most papers are published in *Physics Letters A*, *Int. J. Bifurcation and Chaos*, *Physical Review* Series, *IEEE Trans. on Circuits and System* and *IEEE Int. Symposium on Circuits and Systems*.

In chaotic cryptography, there are two main design paradigms: in the first paradigm chaotic cryptosystems are realized in **analog** circuits (mainly based on chaos synchronization technique)[27], and in the second paradigm chaotic cryptosystems are realized in **digital** circuits or computers and do not depend on chaos synchronization technique. Generally speaking, synchronization based chaotic cryptosystems are generally designed for secure communications though noisy channels and cannot directly extended to design digital ciphers in pure cryptography. What's worse, many cryptanalytic works have shown that most synchronization based chaotic cryptosystems are not secure since it is possible to extract some information on secure chaotic parameters[28–32, 35, 39, 42–44]. Therefore, although chaos synchronization is still actively studied in research of secure communications, the related ideas have less significance for conventional cryptographers. Since this dissertation is devoted to research lying between chaotic cryptography and traditional cryptography, only **digital chaotic ciphers** will be discussed in this dissertation. A comprehensive survey on advances in digital

chaotic ciphers will be given in the next chapter.

§1.2 Original Key Contributions of this Dissertation

Research on digital chaotic ciphers given in this dissertation is initially motivated by our attention on security issues about medical imaging system. When I attended the development of RA3900 II DSA[146] (digital subtraction angiography) imaging system in 1999, security issues were considered in the system to support secure communications of medical imaging through network environment. After that, partially supported by a grant from National Natural Science Foundation of China (No. 30070225) and a grant from National "863" Program (No. 2001AA114152), research on digital chaotic ciphers are carried out and this dissertation is a description of all achievements made by us in this area.

This dissertation involves the following aspects about digital chaotic ciphers: theoretical analyses of dynamical degradation of digital chaotic systems, cryptanalyses of a number of recently-proposed digital chaotic ciphers, and proposals of new digital chaotic ciphers. The first aspect is the basis of the latter two, and the second aspect is another basis of the last aspect. Original key contributions of this dissertation are listed as follows:

1. Because of piecewise linear chaotic maps (PWLCM-s) have perfect dynamical properties and can be realized simply in both hardware and software, they are widely used in digital chaotic ciphers. However, the lack of a theoretical explanation on dynamical degradation of digital chaotic systems makes the analysis of such digital chaotic ciphers difficult. Aiming at piecewise linear chaotic maps, this dissertation discovers a series of measurable dynamical indicators that can quantitatively show the dynamical degradation of the maps in finite precision. The calculation of exact values of the dynamical indicators are studied and some theoretical results are obtained rigorously. The proposed dynamical indicators can be used to compare performances of different remedies to dynamical degradation of digital chaotic systems, and can also be used to find defects in related digital chaotic ciphers and chaotic PRNG-s.

2. Using the proposed dynamical indicators of PWLCM-s, some digital chaotic stream ciphers proposed by Hong Zhou et al. are cryptanalyzed with weak-key analyses. Possible solutions to enhance attacked ciphers are compared and some ones are suggested to enhance security of attacked ciphers.

3. In 1999, E. Alvarez et al. proposed a chaotic cipher based on searching plaintext in a pseudo-random sequence generated from chaotic systems, but soon

it was broken by G. Alvarez et al. in 2000. This dissertation analyzes why E. Alvarez et al.'s cipher is so vulnerable to G. Alvarez et al.'s attacks, and proposes a modified scheme to enhance the security of the original cipher.

4. In 1998, M. S. Baptista proposed a searching based chaotic cipher, which attracted much attention after its proposal. Some cryptanalytic works and modifications are made in recent years. This dissertation points out the deficiency of an attack proposed by Goce Jakimoski and Ljupčo Kocarev, and presents a remedy to resist all known attacks. In the proposed remedy, an interesting feature called probabilistic decryption is found and its further use in cryptology is left for in-depth study in future.

5. In 2001, S. Papadimitriou et al. proposed a probabilistic cipher based on chaotic systems with fast speed. This dissertation analyzes problems of this chaotic cipher and points out its insecurity and impracticalness. Some incorrect results given by S. Papadimitriou et al. are also be rectified.

6. In recent years, J.-C. Yen and J.-I. Guo et al. proposed several chaotic image encryption methods. This dissertation cryptanalyzes two Yen-Guo chaotic image encryption methods (CKBA and BRIE), and proposes known/chosen plaintext attack to break the two systems. Also, some security problems of BRIE is discussed in detail.

7. Based on theoretical results on digital chaotic systems and cryptanalyses of several recently-proposed chaotic ciphers, this dissertation proposes a new chaotic PRBG and uses it to design chaotic stream ciphers with better overall performances. The proposed chaotic PRBG can be used instead of LFSR in conventional stream-cipher cryptography to construct more flexible ciphers.

8. Based on theoretical results on digital chaotic systems and cryptanalyses of several recently-proposed chaotic ciphers, this dissertation proposes a fast chaotic cipher. This cipher is specially designed to fulfill needs of real-time video encryption. Detailed analyses show that the proposed chaotic cipher can provide rather fast encryption speed and high level of security simultaneously. The cipher can also be considered as a general model of new digital (chaotic) ciphers.

§1.3 Organization of this Dissertation

Main contents of this dissertation can be divided into three independent parts, and the three parts correspond to the three aspects involved by this dissertation

(see above). Chap. 2 is a more comprehensive survey of state-of-the-art of digital chaotic ciphers. Chap. 3 is about dynamical degradation of digital chaotic systems. Chap. 4 to Chap. 7 are about cryptanalyses of some recently-proposed chaotic ciphers. Chap. 8 and Chap. 9 are our proposals on new chaotic ciphers. Chap. 10 gives the conclusion and future remarks. A relative detailed introduction of all chapters are as follows:

Chap. 2 gives a comprehensive survey on advances in digital chaotic ciphers from 1980s till now (March 2003). All proposed digital chaotic ciphers are classified into three categories for detailed introduction. As an important issue about digital chaotic ciphers, dynamical degradation of digital chaotic systems in finite computing precision (i.e., in the digital world) is discussed from both theoretical and practical points of view. Other issues in the design of digital chaotic ciphers are also mentioned and my opinion on solutions to these problems are given.

Chap. 3 focuses on theoretical analyses on a series of dynamical indicators of digital piecewise linear chaotic maps (PWLCM-s) and their applications in chaotic cryptography. It is found that dynamical degradation of digital PWLCM-s can be quantitatively measured with the series of dynamical indicators, and the exact values of the indicators have unique relation with the control parameters of PWLCM-s, which are generally selected as the secret key in digital chaotic ciphers based on PWLCM-s. As an independent application, the proposed dynamical indicators are used to compare performance of three remedies on dynamical degradation of digital chaotic systems. The theoretical results obtained in this chapter will be used in Chap. 4 to cryptanalyze some Hong Zhou et al.'s chaotic stream ciphers.

Chap. 4 introduces our weak-key analysis on some Hong Zhou et al.'s chaotic stream ciphers, based on the theoretical results about PWLCM-s proved in Chap. 3. Following the result of weak-key analysis, an enhanced brute force attack is proposed and the key entropy can be reduced 2 bit in a whole. Although the key entropy decreases not much, the proposed attack works well for strongly weak keys and the original ciphers still should be mended. Some possible remedies are discussed and several ones are suggested for practical use and for future study.

Chap. 5 depicts our works on a class of digital chaotic ciphers, in which the ciphertext is generated by searching plaintext in a chaotic pseudo-random sequence. Two typical ciphers were respectively proposed by E. Alvarez et al. in 1999 and by M. S. Baptista in 1998. Some cryptanalyses and modifications are proposed in recent years, and both two ciphers are broken. For E. Alvarez et al.'s cipher, this dissertation discusses why it is insecure and proposes an improved scheme against related attacks. For M. S. Baptista's cipher, this dissertation points

out the deficiency of an attack proposed by Goce Jakimoski and Ljupčo Kocarev in 2001, and presents a remedy to resist this attack and some other new ones (proposed later by G. Alvarez et al. in 2003). In our remedy to M. S. Baptista's cipher, an interesting feature called probabilistic decryption is found.

Chap. 6 is about our cryptanalysis of S. Papadimitriou et al.'s probabilistic chaotic cipher proposed in 2001. It is found this cipher is neither impractical nor infeasible because of its special design. Some wrong statements on this cipher are corrected and the use of this cipher is denied.

Chap. 7 shows two chaotic image encryption methods proposed by J.-C. Yen and J.-I. Guo are not secure to known/chosen plaintext attack. Also, some subtle security defects on one chaotic image encryption method are analyzed. Generally speaking, cryptanalysis given in this chapter can also be extended to break some other image encryption methods proposed by the same authors.

Chap. 8 shows our proposal on a novel chaotic PRBG (pseudo-random bits generator) based on a couple of chaotic systems. It is called CCS-PRBG in this dissertation in short. Both theoretical and experimental analyses show CCS-PRBG has desired cryptographical properties. As examples of CCS-PRBG in stream-cipher cryptography, some stream ciphers based on CCS-PRBG are proposed. It is found that such stream ciphers can reach a better overall performance than other chaotic stream ciphers.

Chap. 9 discusses our proposal on a fast chaotic cipher for real-time video encryption. This cipher is specially designed to reach both high level of security and very fast encryption speed by the following ideas: 1) employing 2^n chaotic systems; 2) combining a chaotic stream sub-cipher and a chaotic block sub-cipher; 3) using time-variant S-box in the chaotic block sub-cipher; 4) using plaintext/ciphertext feedback to make S-boxes also depend on plaintexts. As an impressive result of this cipher, the encryption speed of a prototype system realized in a 700MHz Celeron® CPU reaches about 46 Mbps, which is even faster than software implementations of many conventional ciphers.

Chap. 10 summaries our works in this dissertation and gives some future remarks on future research. Specially, this dissertation gives some useful suggestions on design new digital chaotic ciphers.

Chapter 2
Cryptography Based on Digital Chaos

§2.1 A Historical Overview

To the best of my knowledge, the first paper about ciphers with dynamical systems was Wolfram's paper published in Crypto'85[47], which is a cellular automata based cryptosystem. The second paper [48] is also about cellular automata. But the above both papers did not attract much attention from other researchers. The first paper entitled "chaotic cipher" and frequently cited by following researchers was published by Robert A. J. Matthews in 1989[58], in which a novel stream cipher is suggested based on a generalized logistic map*. From then on, digital chaotic ciphers attract more and more attention of many researchers from different areas[18–22, 24–26, 47–141]. At the same time, cryptanalytic works also have been developed, and some chaotic encryption systems have been found not secure enough[19–21, 39, 53, 59, 64, 66, 70, 88, 97, 100–103, 109, 111, 126–131, 139–141].

After Matthews's paper in 1989, an initial boom of chaotic cryptography lasted for about four years[49, 58–70, 70, 71] chiefly in cryptology field. An interesting mark of the initial boom is that **three** papers [63, 66, 67] appeared in a same conference – EuroCrypt'91 and at least **six** papers [58–60, 64, 65, 71] appeared in a same journal *Cryptologia*. From 1993 to 1996, partially because of the negative results given in [59, 64–66, 70, 71], research in this area becomes rare and only a few results are given[50–52, 72–77]. From 1997, some newly-proposed chaotic ciphers[18–21, 24–26, 53–55, 78–93, 132] open a new splendid boom in the 21^{th} century[22, 38, 39, 56, 57, 94–100, 104–131, 133–141]. In the past few years, some review papers on chaotic cryptography have been published[19–21, 34, 101–103], but many digital chaotic ciphers (including all new contributions made after the year of 2000[22, 38, 39, 56, 57, 94–100, 104–131, 133–141]) are not surveyed. In this chapter, I will try to give a much more comprehensive review on of today's digital chaotic ciphers, related problems and possible solutions. Some remarks on future research in this area will be given in the last chapter of this dissertation.

Basically, there are two general ways to design digital chaotic ciphers: 1) using chaotic systems to generate pseudo-random keystream, which is used to mask the plaintexts; 2) using the plaintext and/or the secret key(s) as the initial conditions and/or control parameters, iterating/counter-iterating chaotic sys-

*In the next year, L. M. Pecora and T. L. Carroll discovered chaos synchronization technique and proposed a secure communication approach via chaos synchronization[27]. It is rather interesting that independent ideas on chaotic cryptography occurred almost simultaneously.

tems multiple times to obtain ciphertext. The first way corresponds to stream ciphers and the second to block ciphers. Besides the above two general ways, in past few years some novel ideas to construct chaotic ciphers have been proposed, such as chaotic S-boxed based block ciphers[55, 105, 108, 112]and searching based chaotic ciphers[84, 90, 104, 110, 113–116, 122, 123, 128]. Compared with research on private-key chaotic cryptography, public-key chaotic cryptography always keeps silent, and to the best of my knowledge, only four such cryptosystems have been reported[45, 48, 49, 72]. Fortunately, the chaotic public-key cryptosystems proposed recently[45] seems to be interesting as a general way to use chaos in public-key cryptography.

This chapter is organized as follows. In §2.2, §2.3 & §2.4, this dissertation will respectively review digital chaotic ciphers from the following three classes: generic chaotic stream ciphers, generic chaotic block ciphers and all other chaotic ciphers. Most ciphers categorized into the third class are rather fresh and some ones seem promising with fruitful results. In §2.5, the following issue is focused: when chaotic systems are realized in finite precision, dynamical degradation will occur and digital chaotic ciphers should handle this serious problem. More problems in the design of digital chaotic ciphers and possible solutions are discussed in §2.6. The last section concludes this chapter.

§2.2 Generic Chaotic Stream Ciphers

§2.2.1 Stream Ciphers Based on Chaotic PRNG-s

Because the chaotic systems can generate "unpredictable" pseudo-random orbits, many researchers have paid their attention on algorithms and performance estimation of PRNG-s (Pseudo-Random Number Generator) based on chaos[17, 24, 58, 61, 67–69, 74, 75, 75, 77–79, 81, 82, 86, 92, 95, 95, 125, 147–165]. For continuous-valued chaotic systems, many chaotic pseudo-random sequences have been proved to have perfect statistical properties.

The kernels of most chaotic stream ciphers are chaotic PRNG-s, whose outputs are the keystreams to mask (generally XOR) the plaintexts. Two chief algorithms generating chaotic pseudo-random numbers are: A1) Extracting partial or all binary bits from the chaotic orbit[24, 58, 74, 75, 81, 82, 86, 95, 125, 165]; A2) Dividing the chaotic interval into m^* parts and labeling each part with a unique number between $0 \sim m-1$, and generating pseudo-random numbers by which part the chaotic orbit arrives in[61, 67–69, 75, 77–79, 92]. Please note that there exist mutual relation between the first two classes algorithms: all PRNG-s in A1 can be regarded as

*Naturally, from the implementation consideration, $m = 2^n$.

special cases in A2, and some PRNG-s in A2[61, 69, 79] can be considered as special cases in A1.

In most stream ciphers based on chaotic PRNG-s, only a single chaotic systems is employed. Many different chaotic systems have been used: Logistic map[69, 74, 99, 125] and its generalized version[58], 2-D Hénon attractor[67, 95], Chebyshev map [75], piecewise linear chaotic maps[22, 24–26, 74, 77–79, 81, 82, 107, 117], and piecewise nonlinear chaotic map[92], p-adic discretized chaotic systems[86], first-order non-uniformly sampling DPLL (Digital Phase-Locked Loop) circuits[61, 68, 163], etc.

As a general way to enhance security (and also encryption speed for some chaotic ciphers), multiple chaotic systems are suggested by several researchers[22, 74, 99]. In [74], the outputs of the chaotic systems are XOR-ed, then mask the plaintexts with XOR operations. The Bernoulli shift and Logistic map are used for demonstration. In [99], the authors proposed a similar stream cipher based on two independent chaotic maps, where one chaotic map is perturbed by ciphertext. In [22], the outputs of two chaotic systems $\{x_1(i)\}, \{x_2(i)\}$ are compared to generate pseudo-random bits $\{k(i)\}$: if $x_1(i) > x_2(i)$, $k(i) = 1$, if $x_1(i) < x_2(i)$, $k(i) = 0$, and if $x_1(i) = x_2(i)$, no output (such a chaotic PRBG is called CCS-PRBG[22]). When some requirements are satisfied, the generated bits sequence have perfect properties. Some ciphers based on CCS-PRBG are given to show its potential applications in stream-cipher cryptography. For details of CCS-PRBG based chaotic stream ciphers, please see Chap. 8.

From the works in [59, 64, 65, 70], it has been known that several chaotic stream ciphers[58, 60, 67] are not secure enough. In [141], we made weak-key analysis on Hong Zhou et al.'s cipher in [24] and find a multi-resolution attack can be used to break it with less complexity than brute force attacks. More details are given in Chap. 4. Further research is wanted to estimate the security of other chaotic stream ciphers.

§2.2.2 Stream Ciphers via Chaotic Inverse System Approach

In [76], mainly motivated by research in chaos synchronization based secure communications, U. Feldmann et al. proposed a general model for the design of chaotic secure communication systems, which is called chaotic inverse system approach. Conceptually speaking, chaotic inverse system approach actually restates the basic encryption model of a general cipher, so it can be used in both analog and digital situations and for both stream ciphers and block ciphers. In fact, many chaotic secure communications can be described by this model. In [21, 83], a general structure of such ciphers (also including some conventional

stream ciphers) is investigated and cryptanalyzed in detail.

Observe the basic structure of inverse system encryption approach, we can see it is more of a general model for chaotic block ciphers than for stream ciphers. Why I classified discussed chaotic ciphers in this subsection into *chaotic stream ciphers*? Actually, all concerned chaotic ciphers in this subsection are more like block ciphers running at CFB (Ciphertext Feedback) mode to generate pseudo-random keystreams (like stream ciphers based on chaotic PRNG-s), which are then used to mask the plaintext with mod1 operation. Here, please note that the plaintext is not mainly encrypted by chaotic system, but by the keystream generated by chaotic systems with feedback from ciphertexts. Thus, I prefer to classify them into chaotic stream ciphers, not chaotic block ciphers, to emphasize the similarity between them and PRNG-s based chaotic stream ciphers. For example, from an algorithmic point of view, we have no reason to think the cipher in [24] (based on chaotic PRNG) is a chaotic stream cipher, but the ciphers in [25, 26] are chaotic block ciphers.

In [25, 26, 73, 98], digital ciphers based on chaotic inverse system approach are presented. They are all stream ciphers with the feedback of the previous ciphertexts: $y(t) = u(t) + f_e(y(t-1), \cdots, y(t-k)) \bmod 1$, where $u(t), y(t)$ represent the plaintext and ciphertext respectively, and $f_e(\cdot)$ is a function generating masking keystream from delayed feedback ciphertexts. In [73], $f_e(t) = a \cdot y(t-1) + b \cdot y(t-2)$; in [25, 26], $f_e(t) = F^m(y(t-1), p)$, where $F(x, p)$ is a piecewise linear chaotic map realized in finite precision $L < m$:

$$F(x,p) = \begin{cases} x/p, & x \in [0,p) \\ (x-p)/(0.5-p), & x \in [p, 0.5] \\ F(1-x,p), & x \in [0.5, 1) \end{cases}. \tag{2.1}$$

The cipher in [98] is actually a variant of the ones in [25, 26] with so-called dual-resolution feature. Assume L is the size of each plain-block, given a secret integer $P \gg 2^L$ and seven secret parameters $p_1, p_2, p_3, c_1, c_2, c_3, c_4$, the encryption procedure can be denoted as: $y(t) = u(t) + \left\lfloor 2^L \cdot F^8_{PWL}(u'(t)) \right\rfloor \bmod 2^L$, where $u'(t) = \left[\sum_{i=1}^{4} \frac{c_i}{P} \cdot y(t-i) \right] \bmod 1$ and

$$F_{PWL}(x) = \begin{cases} x/p_1, & x \in [0, p_1) \\ (x-p_1)/(p_2-p_1), & x \in [p_1, p_2] \\ 1.0-(x-p_2)/(p_3-p_2), & x \in [p_2, p_3] \\ 1.0-(x-p_3)/(1.0-p_3), & x \in [p_3, 1) \end{cases}. \tag{2.2}$$

Apparently, the above equation is a generalized version of (2.1) by cancelling its symmetry to $x = 0.5$. In [111], it is claimed that this cipher cannot resist chosen ci-

phertext attack, the main results given in this cryptanalytic paper seems not right since its basis is wrong: F_{PWL}^8 is simplified to F_{PWL}. However, the use of multiple iterations is a basic reason of p_1, p_2, p_3 are secure, which has been discussed in detail in Hong Zhou et al.'s papers[24–26]. F_{PWL}^m is suggested again in [111] to solve security problem of single iteration of F_{PWL}*.

The chaotic ciphers proposed in [119, 124] are also based on inverse system approach (although the authors did not claimed so). One-way couple map lattices (OCML) serve as the chaotic systems, and multiple maps are simultaneously used for encryption and decryption. The overall performance of the cipher in [124] is claimed better than AES (Advanced Encryption Standard)[166].

The cipher in [73] has been known insecure to the known/chosen-plaintext attack [88], and some security problems in Feldmann's general model have been pointed out by Hong Zhou et al. in [25]. Our recent works shows that Hong Zhou et al.'s ciphers are also not secure from the cryptographical point of view[129] (see Chap. 4). Further works are wanted to judge security of other chaotic ciphers.

§2.3 Generic Chaotic Block Ciphers

§2.3.1 Block Ciphers Based on Inverse (Backwards) Chaotic Systems

The idea of using inverse chaotic system to construct block cipher was firstly proposed by T. Habutsu et al. in [62, 63]. To facilitate the following discussion, I call it HNSM cipher in this thesis, named after the initials of the authors' last names. Given the secret key p and the following tent map $F_p(x)$ and its (random) inverse version $F_p^{-1}(x)$:

$$F_p(x) = \begin{cases} x/p, & x \in [0, p] \\ (1-x)/(1-p), & x \in (p, 1] \end{cases}, \tag{2.3}$$

$$F_p^{-1}(x) = \begin{cases} px, & b = 0 \\ 1-(1-p)x, & b = 1 \end{cases}, \tag{2.4}$$

where b is a random bit distributes uniformly in {0,1}. The cipher encrypts each plaintext block $P \in (0,1)$ as follows: setting the initial condition of F_p^{-1} to be P, the corresponding ciphertext block C is calculated by $C = F_p^{-n}(P)$, where n random bits $b_1 \sim b_n$ are used to determine output of F_p^{-1} in each iteration.

*Under n-bit precision, [111] concludes $m \geq n/2$ is enough to resist attacks. But this result is not enough for the map (2.2) and $m \geq n \cdot \log_2 \left(\max(p_1, p_2 - p_1, p_3 - p_2, 1.0 - p_3)^{-1}\right)$ should be satisfied following the analysis given in [24].

Naturally, the plaintext P can be recovered from the ciphertext by calculating $P = F_p^n(C) = F_p^n(F_p^{-n}(P))$. Because quantization errors exist in the chaotic iterations, much more significant bits are needed for the ciphertext than the plaintext to ensure the correctness of decryption results (considering the sensitivity of chaos to initial conditions).

Because of weaknesses caused by piecewise linearity of tent map and the use of n random bits, E. Biham presented a chosen-ciphertext attack and a known-plaintext attack to break the above HNSM cipher[66]. It is a well-known "evidence" of the insecurity of chaotic cryptography and cited widely in literature[144, 145]. However, we found that the practical complexity of Biham's attack is rather high and the original cipher can be easily enhanced with some simple modifications, such as using nonlinear chaotic maps instead of tent map or introduce perturbation mechanism[131].

Some years later after the proposal of HNSM chaotic cipher, several modified versions have been proposed in [50, 51, 80, 91, 107, 116, 117]. Toggle Cellular Automata and Logistic map are suggested in [50, 51]. Two-dimensional dynamical systems defined on the unit square $[0,1] \times [0,1]$ are suggested in [80], and the one on $[0, L) \times (0, \pi)$ (a chaotic system obeying particle reflection law) in [91]. In [107, 117], a one-to-one chaotic map $\tilde{f}_a$ defined on $\{1/M, 2/M, \cdots, M/M\}$ (called finite-state Baker's map) and its extension $\tilde{F}_A$ on integer space $\{1, 2, \cdots, M\}$ are proposed to construct block ciphers, in which n random bits are avoided since $\tilde{f}_a$ and $\tilde{F}_A$ are one-to-one functions. In [116], multiple 4-segment piecewise linear chaotic maps are used with a finite-length driving key-sequence. At present, it has not been clear whether or not these modified versions are really secure. As a negative result, the cipher proposed in [116] has been successfully broken by G. Alvarez et al.[127], by generalizing Biham's original attacks.

The cellular automata (CA) based ciphers proposed in [50, 51, 55] also employ inverse iterations of a cellular automata for encryption. Because these ciphers employ cellular automata and are not typical block ciphers, I will introduce them in following subsections. In [93], the author employs delayed dynamics to design a block cipher. This cipher can be considered as a variant of HNSM cipher.

§2.3.2 Block Ciphers by Iterating (Forwards) Chaotic Maps

This class of chaotic ciphers have been proposed in [18, 85, 89, 96, 120, 136], mainly as image encryption methods[18, 85, 89, 136].

The ciphers in [18, 85, 89, 136] are based on 2-D chaotic maps. The basic procedure of these ciphers can be described as follows: iterate a 2-D map to pseudo-randomly permutate the pixels in plain-image, use some substitution algorithm

to flatten the histogram of plain-image; repeat the above two procedures for n times to obtain cipher-image. In order to permutate plain-images with different (and finite) resolutions with the employed 2-D chaotic maps, the maps should be defined on a spatially discrete lattice (corresponding to pixels) to make the discretized map become a bijection at any resolution, such as the discretized Baker map in [18], the discrete Kolmogorov Flow in [85] and the truncated Baker transformation in [89, 136]. In fact, the product of the pseudo-random permutation driven by 2-D chaotic maps and the substitution makes cryptanalysis much difficult. Till now, no attack has been reported to break the above three ciphers.

The ciphers in [96, 120] are based on two different cascaded chaotic systems f, g. To ensure the correctness of decrypted data, the periods of both chaotic orbits should be fixed. Assume P_i, C_i respectively represent the i^{th} plaintext and the i^{th} ciphertext, encryption can be denoted by $C_i = g[k, f(n, P_i)]$ and decryption is $C_i = g[K - k, f(N - n, P_i)]$, where N, K are respectively fixed periods of f, g. Although no cryptanalytic works is published to break this cipher, I think it is unreasonable to use fixed period of chaotic orbits*, since it actually makes the decryption speed VERY slow and makes the known/chosen plaintext attacks feasible (N, K cannot be cryptographical large).

§2.4 Other Chaotic Ciphers: Ideas in New Century

In this section, I will try to give a survey of new ideas on digital chaotic cipher proposed near and after year 2000. Introduction to cellular Automata based chaotic ciphers and chaotic public-key ciphers are also placed here to make the description more clearer. Since our cryptanalytic works are mainly made for this class, details on some chaotic ciphers introduced in this section can be found the following chapters of this dissertation.

§2.4.1 Searching-Based Chaotic Ciphers (See also Chap. 5)

In [84, 90], chaotic ciphers based on searching plaintexts in pseudo-random sequences are proposed. In this dissertation, such ciphers are called *searching-based chaotic ciphers*. Because of the special design of the two ciphers, it is somewhat difficult to classify them into stream ciphers or block ciphers: M. S. Baptista's cipher is more of a stream cipher than block cipher, while E. Alvarez et al.'s cipher is more of a block cipher than stream cipher; most modified versions of the two ciphers work like stream ciphers, but some ones like block ciphers.

*As a comparison, almost all other chaotic ciphers try to avoid predictable period as possible as they can.

For M. S. Baptista's cipher in [84], the pseudo-random sequence is the chaotic orbit itself. The encryption procedure can be described as follows: splitting the chaotic interval into S units representing different plaintexts, iterating the chaotic system until the orbit arrives in the unit representing the current plaintext and $C_n > N_0$ and the output of a PRNG $\kappa \geq \eta$ (here $\kappa, \eta \in [0, 1]$), and recording the number of chaotic iterations C_n as the ciphertext. Logistic map is selected for demonstration, but other chaotic systems should be also used. The cipher has features of both stream cipher and block cipher.

For E. Alvarez et al.'s cipher in [90], the sequence is generated with the following threshold algorithm from chaotic orbit $\{x_n\}$: $x_n \leq U \rightarrow 0, x_n > U \rightarrow 1$, where U is a threshold and it can be time-variant. For one plaintext with b_i bits, the cipher runs as follows: arbitrarily select an initial condition of a chaotic system, run the chaotic system and generating a pseudo-random sequence C, search the current plaintext in C until it is found; then record the current state of the chaotic system, the current threshold U_i and b_i as the ciphertext. If the plaintext cannot be found in a long time, then $b_i - -$ and repeat the above procedure until the ciphertext can be generated. The following tent map is used to show the performance of the cipher:

$$F(x) = \begin{cases} rx, & x \in [0, 0.5] \\ r(1-x), & x \in (0.5, 1] \end{cases}. \tag{2.5}$$

Essentially speaking, this cipher is a block cipher with data expansion and time-variant block size.

Just several months after the proposal of E. Alvarez et al.'s cipher, G. Alvarez et al. pointed out that the proposed cipher is rather weak and can be easily broken by four attacks[97] when the above tent map (2.5) is used. In [100], G. Jakimoski & L. Kocarev also independently presented a known-plaintext attack to E. Alvarez et al.'s cipher. In [110], we analyzed why the original E. Alvarez et al.'s cipher is so vulnerable to proposed attacks and suggested an improved scheme to avoid known attacks: select the initial condition and the control parameter(s) of the chaotic system as the secret keys, iterate the chaotic system to generate the pseudo-random sequence C, search the plaintext in C and record the iteration number as the ciphertext. Apparently, this improved scheme is similar to M. S. Baptista's cipher.

M. S. Baptista's chaotic cipher attract much attention after its proposal, and some modified versions are proposed by other researchers. In [104], W.-K. Wong et al. suggested introducing an extra pseudo-random number to flatten the distribution of the ciphertexts. In [114], K.-W. Wong introduced dynamically updated look-up-table to obtain faster encryption speed and enhance the security. In [123]

K.-W. Wong et al. enhanced the updating algorithm of look-up-table and suggested adding a session key to make the ciphertext shorter. As an additional fruit, K.-W. Wong extended the idea of dynamically updating look-up-table to realize hashing simultaneously in [122]. In [113], A. Palacios and H. Juarez suggested using cycling chaos in multiple coupled chaotic maps to enhance original M. S. Baptista's cipher.

In 2002, the same group of [90] proposed two new ciphers in [115, 116], as alternative solutions to the security of their previous cipher. The two major features of the cipher proposed in [115] are: 1) coupled map network is used instead of a single chaotic Logistic map; 2) the ciphertext into the number of iterations, thus becoming a Baptista-type cipher. The new cipher proposed in [116] is actually a chaotic block cipher based on inverse chaotic iterations, and has been known insecure as we discussed in §2.3.1.

In [100], G. Jakimoski & L. Kocarev cryptanalyzed M. S. Baptista's cipher and pointed out that it can be broken by a known-plaintext attack, which actually belongs to one-time-a-pad attacks. More details on Jakimoski-Kocarev attack and other three attacks are discussed in [126] from the viewpoint of symbolic dynamics of employed chaotic systems. Conceptually speaking, all proposed attacks can also be used to break the ciphers in [104, 110, 113, 114, 122, 123], since they adopt similar encryption scheme to the original cipher. But the modified ciphers in [114, 122, 123] with dynamically updated look-up-tables can partially resist symbolic dynamics based attacks proposed in [126], since the dynamical look-up-tables confuse the relation between chaotic orbits and ciphertexts. In [128], we argued that Jakimoski-Kocarev attack is not so effective as they claimed in [100], and a practical countermeasure is proposed to resist the attack. It seems that our countermeasure also can resist G. Alvarez et al.'s attacks based on symbolic dynamics, since the iterating numbers (the ciphertexts in the original cipher) of the chaotic maps are concealed. For more details of our opinions on searching-based chaotic ciphers, please see Chap. 5.

§2.4.2 Constructing S-Boxes with Chaos in Block Ciphers

Compared with other ideas of digital chaotic ciphers, generating S-boxes via digital chaos may be a more promising and essential way to connect chaos with conventional cryptography. There are two classes S-boxes generated from chaos: dynamic S-boxes and fixed S-boxes.

Dynamic S-Boxes from Chaos

The initial idea of dynamic substitution and transposition can be traced to Terry Ritter's early papers [167, 168]. Ritter's methods are rather similar to W.-K. Wong suggested in [114].

To the best of my knowledge, the first idea about generating dynamic S-boxes with chaos is found in a cellular automata based block cipher[55]. The S-boxes are generated from a cellular automata controlled by two initial keys k^{-1} and k^0. At the encryption side, the S-boxes are determined by backwards iterating the CA, and at the decryption side, they are determined by running the CA forward. If we consider the dynamic substitution operations as a masking function like XOR used in common stream ciphers, then the CA based cryptosystem is more of a stream cipher than block cipher.

Also, in [87], a probabilistic block cipher is designed by Donghui Guo et al. with the use of chaotic attractors in neural networks. In this chaotic cipher, a pseudo-random number generator is used control a neural network with a sub-key M to generate time-variant ciphertexts for identical plaintext. Here, the time-variant substitutions from plaintext to ciphertext can be also considered as dynamic S-boxes. An hardware implementation of the above cipher (by the authors themselves) was reported in [94].

In [112], we firstly explicitly suggested the use of dynamic S-boxes generated from chaos. We proposed a fast product cipher containing a chaotic stream sub-cipher and a chaotic block sub-cipher. In total $2^n + 1$ piecewise linear chaotic maps are employed, in which 2^n ones are used for encryption (called ECS – Encryption Chaotic System) and another one is used for controller (called CCS – Control Chaotic System). The initial condition and control parameter of the CCS serve as the secret key. In the stream sub-cipher, the 2^n ECS-es are iterated to generate signals masking plaintexts. In the block sub-cipher, a pseudo-random S-box is dynamically generated by sorting chaotic orbits of the 2^n ECS-es, and then the S-box is used to substitute the plaintexts masked by the stream sub-cipher. Both sub-ciphers are controlled by the CCS. The initial results show that this cipher has rather fast encryption speed, especially when it is realized with hardware. But the original cipher given in [112] is not secure enough and we will enhance it in Chap. 9 with internal feedback or ciphertext feedback. The idea of generating S-boxes by sorting 2^n chaotic orbits can also be extended to construct general chaotic block cipher (like the ones in [105, 108], see the next sub-subsection **Fixed Chaotic S-Boxes from Chaos**), and to design fast chaotic ciphers with considerable security.

In [118] a chaotic block cipher is proposed with similar idea: the tent map (2.3) is iterated to dynamically generate pseudo-random noise vectors and S-

boxes to encrypt plaintexts. This cipher runs in CBC (Cipher Block Chaining) mode[144, 145], and the feedback of ciphertexts makes known/chosen plaintext attacks more difficult.

Finally, please note that the dynamically updated look-up-tables in [114, 122, 123] are actually dynamic S-boxes, but the dynamic updating algorithm is not controlled by chaos. In addition, the algorithm to generate 2^e shuffled integers in [106] is also can be extended to generate dynamic S-box (but the speed is relatively slow).

Fixed Chaotic S-Boxes from Chaos

Following the design criterion of conventional block ciphers, L. Kocarev et al. have suggested construct chaotic block ciphers by introducing chaotic systems to construct S-boxes (nonlinear round functions)[19, 105, 108]. They proposed two algorithms generating S-boxes: a) define a specific discretized one-to-one map from a chaotic map, such as the discrete version of map (4) in [19] and map (12) in [108]; b) iterate a chaotic map to generate a shuffled sequence of 2^n integers $1, 2, \cdots, 2^n$, which can be used as a $n \times n$ S-box*. Recently, their further works have shown that the generated S-boxes can resist differential and linear cryptanalysis[121].

Actually, L. Kocarev et al.'s ideas are methods to design nonlinear S-boxes with acceptable cryptographical properties, not approaches to design new structures of chaotic block ciphers. Therefore, such an idea can be naturally used in conventional block-cipher cryptography. Similarly, many other algorithms can also be used to design S-boxes from digital chaos, which can then be used in any conventional block cipher. Specially, all methods mentioned in the above subsubsection to generate dynamic S-boxes from chaos are suitable to fulfil such a task. We believe that fixed "chaotic" S-boxes will enrich the toolbox of pure cryptographers.

Since most attacks to break weak S-boxes needs a large number of known/chosen plaintexts, dynamic S-boxes can resist various attacks more natural and easier in comparison with fixed S-boxes. What's more, the performance analyses of the chaotic cipher in [112] have implied that dynamic S-boxes can be a promising way to dramatically promote encryption speed of chaotic ciphers. Therefore, I think using dynamic S-boxes to construct chaotic block ciphers is promising to be a primitive in digital chaotic ciphers.

*For the procedure to get the shuffled sequence, please refer to their papers [105, 108].

§2.4.3 A Probabilistic Block Cipher Based on Chaotic Systems (See also Chap. 6)

The proposed cipher in [106] is a probabilistic block cipher. A chaotic system, which is composed of K coupled difference equations with K variables, is used to generate 2^d virtual attractors containing 2^e states $1, 2, \cdots, 2^e$, where $e > d$. Given a permutation matrix $\mathbf{P}_{2^d \times 1}$, the ciphertext is randomly selected from all the states allocated into the $\mathbf{P}[M_C]^{th}$ virtual space, where M_C is the plaintext. Although the authors of [106] stated that their cipher has high security, we have found some serious problems of the ciphers[130]:

- Paradox exists between the practical implementation and high security: the size of the ciphertext and the plaintext (d and e) should be large enough to ensure high security, while it should be small enough to enable practical implementation.
- The value of the number of all possible virtual states is deduced by a wrong way.
- The security analysis given in [106] is inadequate and the security to exhaustive attack is overestimated.
- The merit of fast encryption speed is dependent on the defect about the values of d and e.
- When digital chaotic systems are realized in finite precision, the dynamical degradation will arise and some remedy should be employed to improve it.
- No explicit instructions are given to show how to select the 2^d virtual attractors from the 2^e integers, how to allocate the 2^e virtual states into the 2^d attractors, and how to generate the permutation matrix $\mathbf{P}$.

For details about our analyses and problems of the proposed ciphers, please see Chap. 6.

§2.4.4 Cellular Automata Based Ciphers

A cellular automata (CA) can be viewed as a parallel machine simulating a discrete dynamical system. Although two CA based ciphers[47, 48] occurred much earlier than other chaotic ciphers, there are only a small number of such ciphers in the past 18 years[47–57]. Here I will give a brief introduction on all CA based ciphers known to me. Since the CA based cipher proposed in [55] has been discussed in §2.4.2, here I will simply omit it.

In the first CA based cipher[47], a particular CA (known as rule 30) is used as a PRNG in stream-cipher cryptography, and the secret key is selected as the initial state of the CA. In [48, 49], a revertible non-homogeneous CA is carefully constructed so that another CA (the inverse of the above one) can be found from solving complicated system of equations, which makes encryption and decryption asymmetry and public-key cryptosystems are then designed. In [50, 51] a toggle CA is inversely iterated (just like chaotic systems do in [62, 63]) to encrypt plaintexts, and the decryption is made by forward iterations. As I have mentioned in §2.3.1, Logistic map is also suggested as an alternative dynamical system of CA-s. A detailed analyses and comparison of the above CA based cryptosystems can be found in [51].

From 1994 till now, P. P. Chaudhuri et al. have proposed several CA based cryptosystem[52, 56, 57]. The ciphers suggested in [52] has been known insecure because of its inability to change the key and because the cipher generates a subgroup of *affine group* and not the *alternating group*[53]. In [56], another CA based cipher is proposed, but it either unable to come out from the affine group constraint and so fails to achieve desired level of security. Very recently[57], P. P. Chaudhuri et al. proposed a new CA based cipher and realized it with crypto-hardware. Their theoretical and experimental analyses claimed that the proposed cipher is significantly better than DES and comparable to AES, and its encryption/decryption throughput is higher than both DES and AES. This cipher is too new, and further cryptanalytic works are wanted in future to support the authors's arguments on its performance.

§2.4.5 Chaotic Public-Key Ciphers

As I have mentioned in §2.1, only four chaotic public-key ciphers have been known to us. The two public-key cryptosystems in [48, 49] are both based on cellular automata and have been introduced in the last subsection. The security of Guan's and Kari's CA based public-key cryptosystems has not been clear at present (no cryptanalysis has been reported till now). Here, I would like to focus on another two chaotic public-key ciphers presented in [45, 72].

In Chap. VI of his Ph. D. dissertation[72], Fengi Hwu proposed a chaotic public-key encryption scheme, which is a variant of ElGamal's scheme[144, 145] and can be depicted as follows. Each user selects and publishes (a_0, a_n, α) as his public encryption key and uses n as his secret decryption key, where α uniformly distributes in $\{1, \cdots, p-1\}$ and a_n is calculated by iterating the following digital chaotic map for n times from a_0: $F(x) = \alpha x \bmod p$ or $F(x) = x^2 \bmod p$ (or even more complex map such as $F(x) = x^r + d \bmod p$). The integer p is a large prime

(about 200 digits) so that $p-1$ has a large prime factor and α is a primitive element of p. Now the encryption and decryption procedure can be described as the following steps:

- **Encryption at Sender's Side**: The sender randomly generates a positive secret integer k, and iterates the chaotic system from a_0 for k times to get a_k and a_n to a_{n+k} as well;
- **Ciphertexts in Communication Channel**: The transmitted ciphertext are composed of $c_1 = a_k$ and $c_2 = m \times a_{n+k} \bmod p$;
- **Decryption at Receiver's Side**: The receiver iterates the chaotic system from $c_1 = a_k$ for n times to get $s = a_{n+k}$, and then get the plaintext by $m = (c_2/s) \bmod p$.

The above public-key cryptosystem is almost same to ElGamal's scheme expect that the use of different one-way function. Although Fengi Hwu argued that the security of the proposed scheme is as secure as ElGamal's scheme, its feasibility is problematic. For each secure communication, the sender has to iterate the chaotic system for $2k$ times and the receiver has to iterate it for n times. However, generally k and n are cryptographically large integers, the encryption and decryption speed will become terrifically slow. On the other hand, the attacker can solve n by simply iterating a_0 until he gets a_n, where "only" n iterations are needed (as fast as the receiver to decrypt the ciphertext!). That is to say, if n is too large, then the cryptosystem becomes impractical; but if n is too small, the cryptosystem is not secure at all. Apparently, the problem lies in the fact that multiple iterations of general chaotic maps **cannot be reduced with some efficient techniques** (not like $x^\alpha \bmod p$, which can be dramatically reduced with addition chaining technique widely used in today's public-key cryptography[144, 169]).

Yet another public-key cryptosystem based on chaos is proposed very recently in [45] and is called distributed dynamical encryption (DDE) by the authors. Actually, this cryptosystem works with the following idea: split a dynamical system of dimension $D_T + D_R$ into two parts with D_T transmitter (public) variables and D_R receiver (private) variables. The transmitter sends a scalar signal $s_t(n)$ in which the plaintext $m(n)$ (with two possible value 0 or 1) is embodied to receiver, and the receivers sends another scalar signal $s_r(n)$ back to the transmitter. Given two attractors of the whole dynamical system, one plain-bit $m(n)$ is recovered by at which attractor the system converge after L chaotic iterations. This DDE system is demonstrated with the dynamics of a coupled map lattice. In such a cryptosystem, the positions of the above-mentioned attractors should be altered for each bit to frustrate known plaintext attacks, but such altering will dra-

matically adds the receiver's computation load. Also, when noise occurs in communication channels, the authors analyzed how to resist HMM (Hidden Markov Model) based attacks and reach the conclusion that larger noises may lower security. Although it is possible to find security weaknesses of the proposed system, I still believe that the basic idea used in this public-key cryptosystem opens a new direction of chaotic cryptography.

§2.4.6 Chaotic Image Encryption Methods

There are some chaotic ciphers specially designed for image encryption[18, 85, 89, 132–138]. The chaotic image encryption methods proposed in [18, 85, 89, 136] have been introduced as generic chaotic block ciphers in §2.3.2, and all others[132–135, 137, 138] are proposed by J.-C. Yen and J.-I. Guo (et al.). Yen-Guo chaotic image encryption methods all yield the following basic idea: a chaotic map serves as a chaotic PRNG, and the PRNG is used to control secure permutations or substitutions of pixels. From the cryptographical point of view, all Yen-Guo cryptosystems are not secure since known/chosen plaintext attack can break them with less complexity than brute force attack (some ones can be broken with only several plain-images). Detailed analyses of two chaotic image encryption methods have been attacked by us in [139, 140]. More discussions on Yen-Guo chaotic image encryption methods and our cryptanalyses on the two attacked ones will be given in Chap. 7.

§2.5 Dynamical Properties of Digital Chaos

For digital chaotic ciphers, the employed chaotic systems are realized in digital world, and it is reasonable to doubt they can exactly preserve dynamics of continuous chaotic systems because of the following fact: in classical chaos theory, all dynamical systems are defined in continuum, and their dynamical properties have their meanings only in continuous phase space with positive Lebesgue measure. In this section, let us investigate questions about digital chaos: What will occur when continuous chaos become digital chaos? Do dynamical properties of continuous chaos yet preserve in digital world? What is the influence of such digitization on digital chaotic ciphers? Although the above questions have been considered by only a few researchers[81, 170, 171] in this area, we really think they are VERY important to ensure theoretical security of digital chaotic ciphers. In fact, the lack of careful investigations on dynamical properties of digital chaotic system is the reason of failures of some previous chaotic ciphers[70, 141] and why many conventional cryptographers would not like to believe the security of most

digital chaotic ciphers[145, §3.6].

When we use chaos in digital circuits or computers, the dynamical systems will be discretized both spatially and temporally, that is to say, they will become *discrete-time and discrete-value chaotic systems*[102] defined in discrete time and spatial lattice with finite elements. A natural way to understand such discretized chaotic systems is to consider them as *ϵ-discretized* chaotic systems perturbed by (deterministic) quantization (round-off, truncated or ceiled) errors in discrete iterations[172], where ϵ is the maximal distance between neighboring points in the lattice. In this dissertation, I will only focus my attention on discretization of chaos in digital space (i.e., in 2^{-N}-discretized space, where N is the finite precision of digital arithmetic) and call discretized chaotic systems as *digital chaotic systems*. In this dissertation, digital chaos is also called pseudo chaos[143] and the perturbed chaotic orbits are called *pseudo orbits*[173] to emphasize their essential difference from continuous chaos and continuous orbits.

§2.5.1 Theoretical Aspect: Dynamical Degradation of Digital Chaotic Systems

When using chaos in digital ciphers, many researchers have found dynamical degradation of digital chaotic systems and such degradation threatens security of designed chaotic ciphers[64, 70, 81, 82, 109, 141, 170, 171]. Actually, motivated by "strange" phenomena on chaos obtained in digital computers and numerical experiments, pathologies of digital chaos have been extensively studied in chaos theory[109, 143, 171–205]. To show how such dynamical degradation occurs, assume the discretized space has 2^N finite elements, let us consider the following facts.

Intractable Quantization Errors

The quantization errors, which are introduced into iterations of digital chaotic systems for every iteration, will make pseudo orbits depart from real ones with very complex and uncontrolled manners. Because of the sensitivity of chaotic systems on initial conditions (and also control parameters), the pseudo orbits in finite precision can be expected to be entirely different from the theoretical ones even after a short number of iterations (the lower bound of the number can be calculated from Kolmogorov entropy[206]). In [190], the authors given a good demonstration on this problem: for a piecewise linear chaotic system, when the system is realized respectively in single floating-point precision and in double precision, the obtained pseudo orbits are topologically different and both two are far different from the theoretical one solved from the equations (see Fig. 5 to Fig. 7 of [190]). A good works on the relation between computer arithmetic (floating-

point) and digital dynamical systems can be found in [188], and it has been shown that even "trivial" changes of computer arithmetic can definitely change pseudo orbits' structures.

Although all quantization errors are absolutely deterministic when the finite precision and the arithmetic are fixed, it is technically impossible to know and deal with all errors in digital iterations (just like chaos itself and can be naturally considered as "quantization chaos" since the round-off function is also a nonlinear equation with bounded phase space*). Some random perturbation models have been proposed to depict quantization errors in digital chaotic systems[23, 172, 194], but they cannot exactly predict the actual dynamics of studied digital chaotic systems and has been criticized because of their essentially deficiencies (some counterexamples are given in [186], such as the tent map with $p = 0.5$: $F(x) = 1 - 2|x - 0.5|$).

Since untractable quantization errors can tell us nothing on digital chaotic systems except the existence of "quantization chaos", we would like transfer to investigate long-term dynamics of pseudo orbits, where some useful results have been found by extensive studies.

Long-Term Dynamics: Unavoidable Periodic Pseudo Orbits

Since digital chaotic iterations are constrained in a discrete space with 2^N elements, it is obvious that every chaotic orbit will eventually be periodic[207], i.e., finally go to a cycle with limited length not greater than 2^N.

In Figure 2.1, I give the schematic view of a typical orbit of a digital chaotic system. Generally, each digital chaotic orbit includes two connected parts: $x_0, x_1, \cdots, x_{l-1}$ and $x_l, x_{l+1}, \cdots, x_{l+n}$, which are respectively called *transient (branch)* and *cycle* in this dissertation. Accordingly, l and $n+1$ are respectively called *transient length* and *cycle period*, and $l+n$ is called *orbit length* (please not different terms may be used by different researchers).

Conceptually, there are only a small number of limit cycles for all pseudo-orbits, which means the digital phase space will contrast to an attractor whose size is smaller than 2^N. Apparently, such a collapsed phase space will destroy the ergodicity of the continuous systems. For a simple example, for the tent map $F(x, p)$ given in (2.3), when it is realized in 4-bit finite precision with round-off arithmetic and when $p = 3/2^4$, we can calculate all pseudo-orbits to draw an orbit-graph shown in Figure 2.2.

Then some questions arise: how to estimate the maximal (and mean) tran-

*Of course, this term "quantization chaos" is rather informal and can only have an reasonable analogy with continuous chaos. On the other hand, consider there are many paradoxical definitions of chaos[17], "quantization chaos" may be rigorous in a sense.

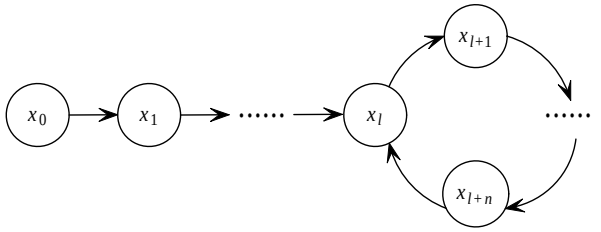


Figure 2.1: A pseudo orbit of a digital chaotic system

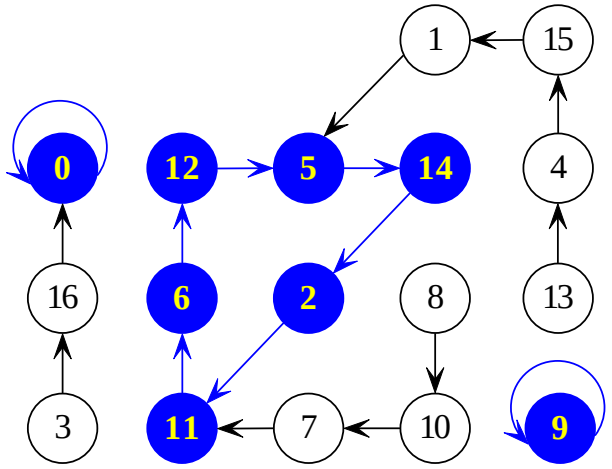


Figure 2.2: The orbit-graph of the digital tent map $F(x, p)$ when $p = 3/2^4$ in 4-bit finite precision (with round-off arithmetic). The node marked with the number i denotes $x = i/2^4$.

sient lengths, cycle periods and the number of limit cycles? Are the lengths large enough to ensure the simulation of the dynamical properties of continuous chaotic systems?

Because of the important role of numerical experiments in studies on chaos, since the establishment of chaos theory, many researchers have made their efforts to answer the above questions[143, 173, 174, 176, 178, 179, 183–186, 191, 192, 196, 201, 204]. Unfortunately, as B. V. Chirkikov and F. Vivaldi reviewed in [143], rigorous studies of such estimations (especially the average lengths) are "notoriously difficult" and the difficulties are actually from the lack of an ergodic theory of discrete chaotic systems. Since the theoretical analysis is not possible, statistical (Monte Carlo) experiments are widely used to explore this issue*. Also, theoretical analyses on

*Some special techniques are also developed to facilitate analyses, such as tree structures proposed in [177] and number theory based (and/or algebra based) tools in [182, 187, 189, 197, 200, 205]. Till now the use of these tools are limited, since they are mainly useful for chaotic systems discretized with special forms, such as p-adic maps and 2-D Hamilton maps.

random mappings[169] serve as reasonable references to predict and confirm the experimental data[173, 174, 198, 203, 204]. Motivated by pioneering works made by F. Rannou[174] and Y. E. Levy[173], an important finding of the above investigated metrics is clarified and confirmed for many different chaotic systems: the scaling law, which seems to also mean fractals of pseudo orbits*. Simply speaking, assume $\epsilon = 2^{-L}$, the scaling law reveals the following facts:

- The maximal (and mean) transient lengths, cycle periods of pseudo orbits both yield $O(\epsilon^{-d})$, where d is a positive indicator uniquely determined by chaotic systems and generally $\epsilon^{-d} \ll 2^L$ (for one-to-one Hamilton maps, $\epsilon^{-d} \ll 2^L$ may be not true[174, 176]).
- The number of limit cycles yields $O(\ln \epsilon^{-1}) = O(L)$.
- The distribution of the cycle period is not uniform, but (roughly) a monotonic decreasing function with respect to the cycle period [184, 185], which means there are a large number of pseudo orbits with short cycle period.

Of course, we should notice that the above results just hold in a general sense and some digital chaotic systems may not yield it. For example, for two typical chaotic maps $F(x) = 1 - 2|x - 0.5|$ and $F(x) = 2x \bmod 1$, the estimations of the *transient lengths* and *cycle periods* is useless since the transient length is always not greater than L and the cycle period is always equal to 1 for any pseudo orbit: $\forall i \geq N, F^i(x) \equiv 0$. So, we have to carefully use the above scaling law in practice, especially in the design of digital chaotic ciphers with high level of security.

Zero Measure of Shadowing Periodic Orbits

The β-shadowing lemma of D. V. Anosov and R. Bowen is frequently used to justify the use of numerical simulations of chaotic systems. The shadowing lemma ensures that there exists an exact chaotic orbit close to the pseudo orbit with only a small error[175, 208]. However, this lemma is useless for digital chaos because of the following fact: although the shadowing orbits really exist, they are *trivial* since they are generally of measure zero. Apparently, zero measure of periodic orbits can be induced from zero measure of discrete space in continuum. Also, discretization of phase space may make unstable chaotic orbits stable so that unstable shadowing orbits in continuous space cannot reflect actual dynamics of digital chaotic systems at all. The inability of β-shadowing lemma related with unstable shadowing chaotic orbits is also discussed in [180, 181].

*In [185], the relation between the scaling law and fractal dimension of the studied attractor is connected.

Extreme examples are still the maps $F(x) = 1 - 2|x - 0.5|$ and $F(x) = 2x \bmod 1$ defined in the unit interval [0,1]. For the two well-known chaotic maps, no any quantization error will be introduced during digital iterations, so each pseudo orbit become exact. However, for any digital decimal with $n \leq N$ significant bits, the orbit will converge at zero after n iterations. As a comparison, for real decimals with infinite significant bits (such decimals distribute densely in [0,1] and have the same Lebesgue measure as the unit interval), the chaotic orbits are infinite and chaotic behavior are ruled by orbits of such decimals.

Weak Dynamics: Ergodicity, Invariant Measure, Lyapunov Exponent, and Even More?

As we have mentioned in the above two sub-subsections, all pseudo orbits are eventually periodic and their cycle lengths may be rather short (although there may really exist long cycles[186]), and the shadowing orbits are of zero measure in continuum. The above facts imply possible collapse of continuous chaos, that is to say, the risk of the loss of ergodicity, mixing, invariant measure and positive Lyapunov exponent, etc. To investigate this risk, some efforts have been made from both theoretical and experimental points of view[171, 174, 175, 179, 184, 186, 194, 196, 203]. Although positive results have been reported, we should notice that such results hold in an average sense[171] and from the experimental point of view.

Additionally, there are some other pathologies on digital chaos, and some one have been theoretically analyzed by M. Blank in [172, 195]. There may exist even more subtle and strange phenomena that wait for us to explore. Although plenty of studies have been made in this area, up till now a mature theory* to measure the dynamical properties of digital chaotic systems exactly has not been established at all. To the best of my knowledge, the most comprehensive and detailed discussion on this issue is M. Blank's book "*Discreteness and Continuity in Problems of Chaotic Dynamics*"[172] (however it is a bit old and does not cover enough sub-topics on digital chaos).

§2.5.2 Practical Aspects: How to Purify Digital Chaos in Applications?

When we use chaos in digital applications, an important issue is how to avoid the dynamical degradation of digital chaotic systems so that the actual performance of the designed digital systems will not be reduced.

*Recently, in [202], an interesting model based on Hamming distance instead of Euclidean distance has been proposed to describe discrete chaos and some digital chaotic systems are studied in it.

At first, let us discuss how we should realize chaotic systems in digital computers. Generally speaking, there are two different ways to reach such a task. In the first way, the formulas of original chaotic systems are still used and approximate value of each iteration is calculated in some computer arithmetic (fixed-point, floating-point, or even others). This way is a formal method to realize continuous chaotic systems in digital world. In the second way, the original formulas will be "trivially"* generalized to a specially-designed discrete space in digital computers so that the constructed digital chaotic systems have equal dynamical properties to the original ones. In most actual applications, digital chaotic systems are constructed in the first way, and most digital chaotic systems constructed in the second way are proposed to simplify investigation of dynamical properties and/or to promote the performance of the concerned digital systems[18, 89, 105, 107, 108, 117, 136]. When using the second way in practice, we suggest designers should be very careful to avoid nontrivial defects caused by the changes on original chaotic systems.

Now let us go to the hard kernel of digital chaos in applications: how to purify the digital chaotic systems to cancel the dynamical degradation? As we have reviewed above, there is no systematic theory in today's chaos theory. Fortunately, some practical remedies have been proposed to solve this difficult issue: using higher finite precision[59, 64], cascading multiple chaotic systems[149], and (pseudo-)randomly perturbing the chaotic systems[81, 82, 99, 170, 190, 195, 199, 203]. All solutions are mainly discussed and used in engineering, and the perturbation-based algorithm has been undertaken much investigations. Our detailed analyses on digital PWLCM-s (see §3.4.1) have shown the perturbation-based solution is better than the other two ones, so I strongly suggest its use in digital chaotic ciphers (almost all our papers employ this method[22, 109, 110, 112]). Although it is obvious that proposers of perturbation algorithm do not know whether or not their algorithm is reasonable from a theoretical point of view, it really is supported by some theorists[190, 195, 203]. In fact, as we mentioned above, random perturbation model of quantization errors have been widely adopted by theorists to study dynamics of digital chaotic systems. The engineering perturbation-based algorithm to improve digital chaos is only a byproduct of random perturbation model in chaos theory. Loosely speaking, perturbation-based algorithm can successfully improve the dynamical degradation of digital chaos to fulfill the requirements in different applications, including digital chaotic ciphers.

However, since there are different perturbing methods with different implementation details[81, 82, 99, 170, 199], not all perturbing methods have equivalent

*Please note that it is actually very difficult to make such changes really "trivial". Here, we directly use this word and do not care much about its formal meanings.

performance. Till now there are three typical perturbation methods: perturbing system variable, perturbing control parameter and perturbing both two[199]. For piecewise linear chaotic maps, we will show that perturbing system variable has better performance than perturbing control parameter (for details see §3.4.1). The combined perturbing algorithm is suggested by us to enhance the security weaknesses of a class of chaotic stream ciphers (see §4.6.6).

Generally speaking, the basic procedure of a perturbation algorithm can be described as follows: run a simple PRNG with uniform distribution in concerned discrete space (in which digital chaotic systems is defined) to generate a small pseudo-random perturbing signal $pt(n)$, which is then used to perturb the current chaotic orbit with XOR or other perturbing functions every $\Delta \geq 1$ iterations. In [81, 170], it has been shown that the length of the chaotic orbit T' can be controlled by the cycle length of the perturbing signal T: $T' = \sigma \cdot \Delta \cdot T$, where σ is a positive integer. If the PRNG generates pseudo-random signals with the maximal length 2^L (assume the perturbing PRNG is realized in the same finite precision as the digital chaotic system), the length of any perturbed chaotic orbit will be $\sigma \cdot \Delta \cdot 2^L$, which is even greater than the size of the discrete space 2^L and should be large enough for most applications.

§2.6 How to Make a Good Chaotic Cipher: More Considerations

Besides dynamical degradation of digital chaos, for digital chaotic ciphers, some other problems should be also carefully considered to avoid possible weaknesses and promote the overall performance of designed ciphers. Such weaknesses include potential insecurity caused by the use of single chaotic system, slow encryption speed, and complex implementation (i.e., high implementation cost), etc. For most digital chaotic ciphers proposed before year 2000, many problems are not settled suitably. Although recently some problems have been noticed by researchers and some practical solutions have been proposed[24, 59, 64, 81, 82, 92, 98, 101–103, 107, 117, 170, 171], there is not yet a comprehensive investigation on these problems and possible solutions. In this section, I would like to discuss some typical problems and suggest some practical solutions from both theoretical and experimental points of view.

§2.6.1 How to Select Chaotic Systems

It is the first question one should consider if he/she want to design a digital chaotic ciphers with chaos. Here, we try to give our answer on this question by

the following sub-questions. Of course, as we discussed in the last section, here we assume selected chaotic systems will be properly realized to avoid dynamical degradation.

Should We Design Ciphers for All Chaotic Systems?

It is desired that a digital chaotic cipher can work well with a large number of chaotic systems and it will be optimal if almost all chaotic systems can be used without loss of security. Such a property is called *chaotic-system-free* property in this dissertation. However, there are several reasons to make chaotic-system-free property infeasible (evidences of the first reasons can be found in [17]):

1. There are several different definitions of chaos, but in-depth investigations on these definitions have shown that no any definition can cover all aspects of well-acknowledged chaotic systems.

2. If more than one dynamical properties of chaotic system are used in digital chaotic ciphers, but the independence of different dynamical properties makes chaotic-system-free properties obscure.

3. Generally different chaotic systems have different dynamical properties (such as different invariant measure), which makes the design of chaotic-system-free ciphers more difficult.

4. Many simple (and typical) chaotic systems can be easily realized in most applications (such as piecewise linear chaotic maps), so that the demand for chaotic-system-free property can be relaxed much.

The above discussion implies that digital chaotic ciphers should be designed based on specific chaotic systems, which is my opinion on this question. Of course, here the word "specific" means specific dynamical properties of selected chaotic systems, not the chaotic systems themselves.

What Chaotic Systems Should We Select?

There are two main considerations on the selection of chaotic systems used in digital chaotic systems: 1) whether or not their (digital) dynamical properties are desired to ensure the security of designed ciphers; 2) whether or not their implementations are simple enough for most applications to save costs and reach fast encryption speed. The second consideration means the simpler the chaotic system is, the better the overall performance of the cipher will be. More details on the second considerations will be discussed in following contexts, and here

we would like focus on the first consideration. The first consideration has twofold meanings: whether or not dynamical properties of selected chaotic systems fit the requirements of the designed ciphers; whether or not the dynamical properties of selected chaotic systems are essentially stable for different control parameters. For example, if a chaotic cipher depends on ergodicity, then the chaotic system should be ergodic for all control parameters.

Recall our review on state-of-the-art of today's digital chaotic ciphers in §2.2 to §2.4, many chaotic systems with clear chaotic properties and simple formulas are selected by different designers: piecewise linear chaotic maps[22, 24–26, 62, 63, 74, 77–79, 81, 82, 90, 96, 98, 106, 107, 110, 112, 116–118, 120] (PWLCM-s), Logistic map[51, 69, 74, 80, 84, 99, 104, 105, 108, 113–115, 119, 122–125, 132, 134, 135, 137, 138] and its generalized version[58], cellular automata[47–57], 2-D Baker map[18, 89, 136], 2-D Hénon map[67, 95], chaotic neural network[87, 94, 133], coupled map lattice/network[45, 106, 115, 119, 124], piecewise nonlinear chaotic maps[80, 92], quasi-chaotic digital filters[73], etc. Unsurprisingly, the most focused systems are two well-studied maps in chaos theory: Logistic map $F(x) = rx(1-x)$ and piecewise linear chaotic maps.

The use of Logistic map is apparently because it is the most well-known chaotic system showing complex behaviors[14, 15, 209] and one of the simplest chaotic system. But Logistic map has the following weaknesses for the use in cryptography: 1) its invariant density is not uniform, so that the generated orbits cannot satisfy the balance property of a good cipher; 2) only when $r = 4$, the map is a surjective function on the unit interval [0,1] and exhibits perfect chaotic behaviors; the dynamical properties of Logistic map are different if the control parameter r is different, which may be used by an eavesdropper to collect useful information to lessen attack complexity[97]. As a result, I do not suggest using Logistic map to design digital chaotic ciphers with high level of security.

The use of PWLCM-s is based on their perfect properties[210]: 1) uniform invariant density function; 2) exactness, mixing and ergodicity; 3) exponentially decayed correlation function; 4) simple realization by both hardware and software (like Logistic map). Here, please note the above properties are only true for partial PWLCM-s. In [90], the PWLCM (2.5) is used, while such a PWLCM satisfies the above properties only when $r = 4$ (just like Logistic map, in fact they are topologically conjugate[209]). In addition, even for PWLCM-s who strictly satisfy the above properties, we have proved there still exist measurable dynamical degradation (see Chap. 3). Fortunately, by using some practical remedies, we can avoid weaknesses of PWLCM-s. To sum up, with careful considerations*,

*Another security risk about PWLCM-s is the piecewise linearity, which may cause linear attacks possible if digital ciphers are not designed with careful considerations[66].

PWLCM-s are still good candidates for the design of digital chaotic ciphers with satisfactory overall performances.

Another candidate chaotic system for digital chaotic ciphers is the piecewise nonlinear chaotic map proposed by Tao Sang et al. in [92]. This map has similar dynamical properties to PWLCM-s and can avoid potential security problems caused by piecewise linearity, but floating-point arithmetic is needed to calculate square roots in chaotic iterations, which is not desired to reach fast encryption speed and low implementation cost. I suggest using it only when usability becomes trivial and floating-point computing component is available.

Should We Use One or Multiple Chaotic Systems?

Answer to this question is different for different demands in actual applications. Loosely speaking, the use of multiple chaotic systems will be helpful to enhance security, and promote encryption speed in hardware implementations. But in software implementations, to reach fast encryption speed, less chaotic system is better. Extra discussion can be found in following subsections.

Here, we would like to explain how the use of multiple chaotic systems can enhance security of digital chaotic systems. As we know, almost all digital chaotic ciphers are claimed to be secure by the authors when they are proposed, but many of them are actually not. The reason often lies in the following fact: for many chaotic ciphers, the ciphertext has tight relation with the chaotic orbits, so that intelligent attackers can catch some useful information on system variables and/or control parameters from ciphertext with theoretical tools extracting such information from chaotic orbits[28–32, 35, 109]. As a suggestion, using **multiple** chaotic systems instead of a single chaotic system may be useful to enhance such potential insecurity, since mixing of multiple chaotic systems "should" make cryptanalysis much more difficult, especially when these chaotic systems have different initial condition (and control parameters) and/or different equation. Such an idea has been used in some chaotic cipher[22, 45, 74, 99, 106, 112, 119, 124] to obtain higher level of security, and both theoretical analyses and experiments imply even a couple of chaotic systems are enough to provide acceptable security against information leaking from ciphertext[22].

§2.6.2 How to Reach Fast Encryption Speed?

It is strange that many researchers omitted this issue in their digital chaotic ciphers and cause rather slow encryption speed. As an example, consider the original M. S. Baptista's cipher in [84], at least $N_0 = 250$ chaotic iterations are needed for each plaintext, which makes the cipher runs very very slow. However, it is

absolutely true that any new chaotic ciphers (even with perfect cryptographic properties) will be useless if they can only run with limited speed, since in conventional cryptography there have been so many good ciphers with both high security and fast speed[144, 145]. Recently, some chaotic ciphers with fast encryption speed have been proposed[22, 106, 112, 123]. Actually, the encryption speed of some previous chaotic ciphers can also be optimized with careful remedies. In this subsection, we investigate reasons of low encryption speed of most chaotic ciphers and give some basic principles to promote encryption speed.

Investigate currently known digital chaotic ciphers, we can find the following facts about the encryption speed:

- Many ciphers (such as [18, 24, 25, 45, 50, 51, 61–63, 69, 74, 80, 84, 85, 87, 89–91, 94, 96, 98, 104, 107, 110, 113–117, 120, 122, 125, 165]) use multiple chaotic iterations to generate one ciphertext, which will dramatically reduce the encryption speed. Because chaotic stream ciphers generally needs only one chaotic iterations for each plaintext, their encryption speed is much faster than chaotic block ciphers.

- The encryption speed of the chaotic stream ciphers are mainly determined by the time consuming on chaotic iterations. Consequently, the simpler the chaotic system is, the faster the encryption speed will be. Apparently, PWLCM-s are ones of the simplest chaotic systems, since only one or two multiplications/divisions and several additions/comparisons are needed for each digital chaotic iteration. It is another reason we suggest PWLCM-s in digital chaotic ciphers.

- Some ciphers[84, 90, 104, 110, 113–115, 122, 125] have time-variant speed, so they cannot encrypt plaintexts with fixed bits rate.

- Since the floating-point arithmetic is much slower than the fixed-point one, we suggest using fixed-point arithmetic as possible. So we should avoid using chaotic systems[19, 58, 75, 80, 92, 113] defined by some complicated functions that must calculated with floating-point arithmetic.

- The capability of parallel computation in hardware makes the hardware implementations of digital chaotic ciphers generally much faster than software implementations. So it will better if digital chaotic ciphers contain some parallel computing. For example, when coupled map lattice (or cellular automata) serves as the selected chaotic system in a digital chaotic cipher, the encryption speed will become much faster[119, 124].

§2.6.3 Implementation Issues

Simple implementation by hardware and software at low cost is a very important requirement for a good digital cipher. In fact, implementation problems are crucial factors to influence the use of a cipher in many final applications, since so many ciphers can provide enough security with considerable costs.

The following facts about implementation should be concerned in the design of a digital chaotic cipher (some ones have been discussed above and are emphasize again from the implementation point of view):

- The simpler the employed chaotic system is, the simpler the realization will be and the smaller the cost will be. Now we confirm again that PWLCM-s are the best candidates to design digital chaotic ciphers.
- The fixed-point arithmetic is better than the floating-point one since the latter needs much more cost and computation complexity (not only from viewpoint of encryption speed, as mentioned above).
- For hardware implementations supporting parallel computation, (coupled or independent) multiple chaotic systems will be useful to promote the encryption speed dramatically and add complexity of possible attacks.
- Another desired requirement is the extensible security and accessional functions with considerably extra cost and complexity. The examples of such ciphers can be found in [22, 112].

§2.7 Conclusion

Digital chaotic systems may be a new source of new ciphers, because some dynamical properties can be used to realize the cryptographic properties of good ciphers. In this chapter, we give a comprehensive review of the progress in chaotic cryptography from 1980s till now (2003). Most known chaotic ciphers are classified, discussed and compared. Some problems in the design of chaotic ciphers are detailedly analyzed, and some possible solutions are given. Consider some new chaotic ciphers can provide perfect cryptographic properties, we believe that the chaotic cryptography will be helpful to understand the essence of chaos and also security, and enrich knowledge in cryptology.

Chapter 3
A Series of Measurable Dynamical Indicators of Digital Piecewise Linear Chaotic Maps

§3.1 Introduction

As we have surveyed in Chap. 2, the idea of using digital chaotic systems to construct cryptosystems have been extensively studied since 1980s, and attract more and more attention in recent years. In §2.5, we have shown that digital chaotic systems have complex dynamical degradation, and theoretical analysis of such degradation plays very important role in the design of digital chaotic ciphers with high level of security. Although some coarse measures of dynamical properties of digital chaotic systems have been identified (such as the bounds of periods of pseudo orbits in quantitative order), there are still lack of exact measurable dynamical indicators of specific digital chaotic systems. However, as we suggested in §2.6.1, digital chaotic ciphers should be designed for specific chaotic maps, so the lack of measurable indicators makes the theoretical analysis obscure. In addition, in §2.5.2, we have discussed that digital (pseudo-)random perturbation algorithm can effectively improve the dynamical degradation of digital chaotic systems, but careless use of pseudo-perturbation (deterministic perturbation in actual applications) may still bring subtle security defects.

In this chapter, aiming at digital one-dimensional PWLCM-s, we will introduce a series of measurable dynamical indicators, which can quantitatively reflect dynamical degradation of digital PWLCM-s with different control parameters. Loosely speaking, the studied dynamical indicators are defined as follows. Assume a digital PWLCM $F(\cdot)$ is realized in n-bit finite precision (fixed-point arithmetic is adopted). Given a discrete random variable x distributing uniformly in the discrete space, we have n dynamical indicators defined by $P_j = P\{\text{The least } j \text{ bits of } F(x) \text{ are all zeros}\}(j = 1 \sim n)$. Surprisingly, we found out the following "strange" fact for some PWLCM-s (such as tent map): values of $P_1 \sim P_n$ are uniquely determined by *resolutions* (see §3.2.2 for its formal definition) of slopes of all linear segments (not their concrete values). When we plot the values for different slopes, a regular pattern appears. For general PWLCM-s, the above findings can be qualitatively generalized.

The dynamical indicators can be considered as statistical measures of pseudo-ergodicity of digital chaotic PWLCM-s, and also an evidence of measurable discrepancy of digital invariant measure from its continuous counterpart.

Essentially speaking, these indicators reflect the collapse of digital (fixed-point) divisions on each linear segment and accumulation of such collapse of multiple linear segments. As a natural result, such collapse of digital arithmetic further causes collapse of dynamics of digital PWLCM-s. It can be predicted that such collapse of digital arithmetic should exist for other digital chaotic systems and for other digital arithmetic (such as floating-point arithmetic), and more unseen phenomena lying between continuous chaos and digital computers wait for future exploration. Since this chapter introduces a new systematic method to analyze digital chaotic systems from an arithmetic point of view, hope more fruits can be obtained following such a way*.

Based on the proposed measurable indicators of digital PWLCM-s, we have made a qualitative comparison of different remedies to dynamical degradation of digital PWLCM-s: using higher finite precision, cascading multiple chaotic systems and the perturbation-based algorithm. The results confirm previous investigations from theory of random perturbation model and experiments: (pseudo-)random perturbation may be a better solution to dynamical degradation. What's more, our comparison reveals another fact about perturbation algorithm: perturbing system variables has better performance than perturbing control parameters, which is hardly observed only from experiments. In addition, applications of these measurable indicators in chaotic cryptography and chaotic PRNG-s are discussed in detail. It is found that such measurable indicators can be used to discover black holes hidden behind some digital chaotic ciphers, such as the Hong Zhou et al.'s chaotic stream ciphers[24–26] (in which perturbation is casually adopted to enhance dynamical degradation of digital PWLCM-s, see Chap. 4 for more details)†. All discussions around the proposed dynamical indicators emphasize the significance of theoretical tools in the study on chaotic systems in the digital world.

This chapter is an extension of our paper [109]. In this previous paper, we just strictly proved results on the 1D PWLCM (2.1) and extend the proofs to skew tent map (2.3). This chapter will generalize our theoretical methods given in [109] to make exactly calculating dynamical indicators of all PWLCM-s possible.

This chapter is organized as follows. In §3.2, we firstly give some preliminary knowledge on PWLCM-s, preliminary definitions, some lemmas and corollaries. §3.3 introduces the definitions of studied dynamical indicators and mainly focuses on mathematical proofs of some theorems on how to calculate exact values

*However, similar studies on chaotic maps whose iterations require floating-point arithmetic will be much difficult than fixed-point. It will be helpful if some new theoretical tools are created to model computing procedure of concerned floating-point functions.

†Actually, it is statistical studies on Hong Zhou et al.'s cipher in [24] to make me find the above-mentioned "strange" phenomenon and motivate me to propose the series of dynamical indicators.

of these dynamical indicators for digital 1D PWLCM-s, the two PWLCM-s (2.1) and (2.3) are given as examples to show mathematical meanings of values of all dynamical indicators. In §3.4, we compare performances of three proposed remedies to enhance dynamical degradation of digital chaotic systems, and explain their roles in cryptography and pseudo-random numbers generation. The last section concludes this chapter and gives some remarks on future research.

§3.2 Preliminary Knowledge

§3.2.1 1D Piecewise Linear Chaotic Maps (PWLCM-s)

Just as its name implies, a piecewise linear map (PWLM) is a map composed of multiple linear segments (limited breaking points are allowed). A typical example of PWLM is the tent map (2.3). Because not all PWLM-s can exhibit chaotic behavior, in this chapter I would like to draw my attention on a class of piecewise linear chaotic maps (PWLCM-s) who have good dynamical properties. Another reason of my focus on this class of PWLCM-s is that chaotic maps used in many digital chaotic ciphers[22, 24–26, 62, 63, 74, 77–79, 81, 82, 90, 96, 98, 106, 107, 110, 112, 116–118, 120] belong to this class. However, please note that main results obtained in following contexts are also suitable for other PWLCM-s.

Generally, given a real interval $X = [\alpha, \beta] \subset \mathbb{R}$, let us consider the following PWLM $F : X \to X$:

$$i = 1 \sim m, F(x)|C_i = F_i(x) = a_i x + b_i, \tag{3.1}$$

where $\{C_i\}_{i=1}^m$ is a partition of X, which satisfies $\bigcup_{i=1}^m C_i = X$ and $C_i \cap C_j = \varnothing, \forall i \neq j$. We say the above PWLM satisfies piecewise *onto* property if each linear segment is mapped onto X by F_i: $\forall i = 1 \sim m, F_i(C_i) = X$. If $X = [0, 1]$, it is called a *normalized* 1D PWLM. Obviously, any 1D PWLM can be transformed into its normalized version by simple linear operations:

$$F_{[0,1]}(x) = \frac{F\left(\frac{x-\alpha}{\beta-\alpha}\right) - \alpha}{\beta - \alpha}. \tag{3.2}$$

Apparently, original 1D PWLM is topologically conjugate to its normalized form.

A 1D PWLM with piecewise *onto* property is generally chaotic since it has the following properties on its definition interval X: 1) Its Lyapunov exponent $\lambda = -\sum_{i=1}^m \mu(C_i) \cdot \ln \mu(C_i)$ and satisfies $0 < \lambda < \ln m$, where $\mu(C_i) = ||C_i||/(\beta - \alpha)$; 2) It is exact, mixing and ergodic; 3) It has uniform invariant density function $f(x) = 1/\|X\| = 1/(\beta - \alpha)$; 4) Its auto-correlation function $\tau(n) = \frac{1}{\sigma^2} \lim_{N\to\infty} \frac{1}{N} \sum_{i=0}^{N-1} (x_i - \bar{x})(x_{i+n} - \bar{x})$ will go to zero as $n \to \infty$,

where $\bar{x}, \sigma$ are the mean value and the variance of x respectively; especially, if $\sum_{i=1}^{m} \text{sign}(a_i) \cdot \|C_i\|^2 = 0$ is satisfied, $\tau(n) = \delta(n)$. Property 1,3,4 can be derived with the similar deduction made in [210], and Property 2 holds because $\forall x \in X, |F'(x)| = |a_i| > 1$ except m conjoint/breaking points between two neighbor segments[23]. In the following, without loss of generality, we will use the term PWLCM to represent the above chaotic PWLM.

As we know[23, 206], uniform invariant density function (Property 3) means that uniform input will generate uniform output, and that the chaotic orbit from almost every initial condition will lead to the same uniform distribution $f(x) = 1/(\beta - \alpha)$. However, the above facts are not true for digital chaotic maps. Assume a 1D PWLCM is realized in a discrete space with 2^n finite states. Take 2^n different states as the inputs of the chaotic map, the number of different outputs after one digital chaotic iteration will be smaller than 2^n since any 1D PWLCM is a multi-to-one map. That is to say, for a digital 1D PWLCM, **discrete uniform input can not generate discrete uniform output**, or **a uniform random variable will become nonuniform after digital chaotic iterations**. In this chapter, we try to investigate such an issue: can we exactly measure non-uniformity of chaotic output of a digital 1D PWLCM with (discrete) uniform input? Considering any 1D PWLCM has its conjugate normalized version, we will only focus on normalized 1D PWLCM-s to simplify theoretical analyses.

In order to facilitate the descriptions and proofs of the statistical properties in following sections, we will introduce some preliminary definitions and useful results. In §3.2.2, we give some definitions to depict the discrete space of the real interval $X = [0, 1]$ in n-bit finite precision (Definition 1, 2, 3), and the arithmetic operations on it (Definition 4, 5). In §3.2.3, we give some preliminary results about the definitions introduced in §3.2.2.

§3.2.2 Preliminary Definitions

Definition 3.1: *A discrete set* $S_n = \{a | a = \sum_{i=1}^{n} a_i \cdot 2^{-i}, a_i \in \{0, 1\}\}$ *is called a* ***digital set*** *with* ***resolution*** n. $\forall i < j$, S_i *is called the* ***digital subset*** *with* ***resolution*** i *of* S_j. *Specially, define* $S_0 = \{0\}, S_\infty = [0, 1]$.

We have $\{0\} = S_0 \subset S_1 \subset \cdots \subset S_i \subset \cdots \subset S_\infty = [0, 1]$.

Definition 3.2: *Define* $V_i = S_i - S_{i-1} (i \geq 1)$ *and* $V_0 = S_0$. V_i *is called a* ***digital layer*** *with* ***resolution*** i. $\forall p \in V_i$, i *is called the* ***resolution*** *of* p. *The partition of* S_n, $\{V_i\}_{i=0}^{n}$, *is called the* ***complete multi-resolution decomposition*** *of* S_n; $\{V_i\}_{i=0}^{\infty}$ *is called the* ***complete multi-resolution decomposition*** *of* $S_\infty = [0, 1]$. *For* S_n, *its resolution* n *is also called* ***decomposition level***.

We have $\bigcup_{i=0}^{n} V_i = S_n$, $V_i \cap V_j = \varnothing (\forall i \neq j)$ and $\|V_i\| = 2^{i-1} (\forall i \geq 1)$, where $\|V_i\|$ is the size of V_i. The resolution of a binary decimal $p \in V_i$ is actually the position of its last non-zero bits in the binary representation, i.e., $p = 0.b_1 b_2 \cdots b_i \overbrace{0 \cdots 0}^{n-i} (b_i \neq 0)$. That is to say, **the resolution of p is its binary finite precision**.

Definition 3.3: *$\forall n > m$, $D_{n,m} = S_n - S_m$ is called the **digital difference set** of S_n and S_m (or with parameters n and m). When $m = 0$, $D_{n,0}$ can be briefly written as D_n. $\{V_i\}_{i=m}^{n}$ is called the **complete multi-resolution decomposition** of $D_{n,m}$, and $n - m$ is called the **decomposition level**.*

Definition 3.4: *A function $G : \mathbb{R} \to \mathbb{Z}$ is called an **approximate transformation function (ATF)**, if $\forall x \in \mathbb{R}$, $|G(x) - x| < 1$. Three basic ATF-s are: 1) $\lfloor x \rfloor$ – **floor** function, the maximal integer not greater than x; 2) $\lceil x \rceil$ – **ceil** function, the minimal integer not less than x; 3) $\mathbf{round}(x)$ – **roundoff** function, the rounded integer of x. $\forall x \in \mathbb{R}$, define its **decimal part** as $\mathbf{dec}(x) = x - \lfloor x \rfloor$.*

The above **three** ATF-s (please note **not all** ATF-s) have the following useful properties, whose proofs are rather simple and so we omit them here.

ATF Property 1: $\forall m \in \mathbb{Z}, G(x + m) = G(x) + m$;

ATF Property 2: $a < x < b \Rightarrow \lfloor x \rfloor \leq G(x) \leq \lceil x \rceil$.

Definition 3.5: *A function $G_n : S_\infty \to S_n$ is called a **digital approximate transformation function (DATF)*** with **resolution** n, if $\forall x \in S_\infty = [0, 1]$, $|G_n(x) - x| < 1/2^n$. The following three DATF-s are concerned in this dissertation (also the most frequently adopted DATF-s in digital algorithms): 1) $\mathbf{floor}_n(x) = \lfloor x \cdot 2^n \rfloor / 2^n$; 2) $\mathbf{ceil}_n(x) = \lceil x \cdot 2^n \rceil / 2^n$; 3) $\mathbf{round}_n(x) = \mathrm{round}(x \cdot 2^n)/2^n$.*

The above **three** DATF-s (also **not all** DATF-s) have the following useful properties, which can be easily derived from **ATF Property 1–2**.

DATF Property 1: $\forall m \in \mathbb{Z}, G_n(x + m/2^n) = G_n(x) + m/2^n$;

DATF Property 2: $a < x < b \Rightarrow \mathrm{floor}_n(a) \leq G_n(x) \leq \mathrm{ceil}_n(b)$.

§3.2.3 Preliminary Lemmas and Corollaries

For the three basic ATF-s – $\lfloor \cdot \rfloor$, $\lceil \cdot \rceil$ and $\mathrm{round}(\cdot)$, we have the following Lemma 3.1 and Corollary 3.1, which are useful for the proofs of the theorems in the next section.

Lemma 3.1: *$\forall n \in \mathbb{Z}^+, a \geq 0$, the following facts are true:*

*Following M. Blank's term [172, Chapter 5], G_n is called an *operator of 2^{-n}-discretization*. In this dissertation, to make description simple, we prefer to use the term DATF.

1. $n \cdot \lfloor a \rfloor \leq \lfloor n \cdot a \rfloor \leq n \cdot \lfloor a \rfloor + (n-1)$, *and* $n \cdot \lfloor a \rfloor = \lfloor n \cdot a \rfloor$ *if and only if* $\mathrm{dec}(a) \in \left[0, \frac{1}{n}\right)$;

2. $n \cdot \lceil a \rceil - (n-1) \leq \lceil n \cdot a \rceil \leq n \cdot \lceil a \rceil$, *and* $n \cdot \lceil a \rceil - (n-1) = \lceil n \cdot a \rceil$ *if and only if* $\mathrm{dec}(a) \in \left(1 - \frac{1}{n}, 1\right) \bigcup \{0\}$;

3. $n \cdot \mathrm{round}(a) - \lfloor n/2 \rfloor \leq \mathrm{round}(n \cdot a) \leq n \cdot \mathrm{round}(a) + \lfloor n/2 \rfloor$, *and* $n \cdot \mathrm{round}(a) - \lfloor n/2 \rfloor = \mathrm{round}(n \cdot a)$ *if and only if* $\mathrm{dec}(a) \in \left[0, \frac{1}{2n}\right) \bigcup \left[1 - \frac{1}{2n}, 1\right)$.

Proof: We prove the three sub-lemmas separately:

1. Because $a = \lfloor a \rfloor + \mathrm{dec}(a)$, $n \cdot a = n \cdot \lfloor a \rfloor + n \cdot \mathrm{dec}(a)$. Considering $0 \leq \mathrm{dec}(a) < 1, 0 \leq n \cdot \mathrm{dec}(a) < n \Rightarrow 0 \leq \lfloor n \cdot \mathrm{dec}(a) \rfloor \leq n-1$. From the definition of $\lfloor \cdot \rfloor$, we can get $\lfloor n \cdot a \rfloor = \lfloor n \cdot (\lfloor a \rfloor + \mathrm{dec}(a)) \rfloor = n \cdot \lfloor a \rfloor + \lfloor n \cdot \mathrm{dec}(a) \rfloor \Rightarrow n \cdot \lfloor a \rfloor \leq \lfloor n \cdot a \rfloor \leq n \cdot \lfloor a \rfloor + (n-1)$, where $n \cdot \lfloor a \rfloor = \lfloor n \cdot a \rfloor \Leftrightarrow \lfloor n \cdot \mathrm{dec}(a) \rfloor = 0$, that is to say, $0 \leq n \cdot \mathrm{dec}(a) < 1 \Leftrightarrow \mathrm{dec}(a) \in \left[0, \frac{1}{n}\right)$.

2. i) When $\mathrm{dec}(a) = 0$: $\lceil n \cdot a \rceil = n \cdot a = n \cdot \lceil a \rceil$; ii) When $\mathrm{dec}(a) \in (0,1)$: Assume $\mathrm{dec}'(a) = 1 - \mathrm{dec}(a) \in (0,1)$, then $a = \lceil a \rceil - \mathrm{dec}'(a)$, then $n \cdot a = n \cdot \lceil a \rceil - n \cdot \mathrm{dec}'(a)$. Considering $0 < n \cdot \mathrm{dec}'(a) < n$, $n \cdot \lceil a \rceil - n < n \cdot a = n \cdot \lceil a \rceil - n \cdot \mathrm{dec}'(a) < n \cdot \lceil a \rceil$. From the definition of $\lceil \cdot \rceil$, we can get $n \cdot \lceil a \rceil - (n-1) \leq \lceil n \cdot a \rceil \leq n \cdot \lceil a \rceil$, where $n \cdot \lceil a \rceil = \lceil n \cdot a \rceil \Leftrightarrow n \cdot \mathrm{dec}'(a) \in (0,1)$, then $\mathrm{dec}(a) \in (1 - \frac{1}{n}, 1)$. As a whole, we have $n \cdot \lceil a \rceil - (n-1) \leq \lceil n \cdot a \rceil \leq n \cdot \lceil a \rceil$, and $n \cdot \lceil a \rceil = \lceil n \cdot a \rceil$ if and only if $\mathrm{dec}(a) \in \left(1 - \frac{1}{n}, 1\right) \bigcup \{0\}$.

3. From the definition of $\mathrm{round}(\cdot)$, we have $\mathrm{round}(a) - 1/2 \leq a \leq \mathrm{round}(a) + 1/2$. Thus $n \cdot \mathrm{round}(a) - n/2 \leq n \cdot a < n \cdot \mathrm{round}(a) + n/2$. i) When n is an even integer, it is obvious that $n \cdot \mathrm{round}(a) - n/2 \leq \mathrm{round}(n \cdot a) < n \cdot \mathrm{round}(a) + n/2$. ii) When n is an odd integer, $n \cdot \mathrm{round}(a) - n/2 + 1/2 \leq \mathrm{round}(n \cdot a) < n \cdot \mathrm{round}(a) + n/2 - 1/2$, that is to say, $n \cdot \mathrm{round}(a) - (n-1)/2 \leq \mathrm{round}(n \cdot a) < n \cdot \mathrm{round}(a) + (n-1)/2$. As a whole, we can deduce: $n \cdot \mathrm{round}(a) - \lfloor n/2 \rfloor \leq \mathrm{round}(n \cdot a) \leq n \cdot \mathrm{round}(a) + \lfloor n/2 \rfloor$, where $n \cdot \mathrm{round}(a) = \mathrm{round}(n \cdot a) \Leftrightarrow n \cdot \mathrm{round}(a) - 1/2 \leq n \cdot a < n \cdot \mathrm{round}(a) + 1/2$, that is to say, $\mathrm{dec}(a) \in \left[0, \frac{1}{2n}\right) \bigcup \left[1 - \frac{1}{2n}, 1\right)$.

The proof is complete. ■

Corollary 3.1: $\forall n \in \mathbb{Z}^+, a \geq 0$, *the following facts are true:*

1. $\lfloor n \cdot a \rfloor \equiv 0 \pmod{n}$ *if and only if* $\mathrm{dec}(a) \in \left[0, \frac{1}{n}\right)$;

2. $\lceil n \cdot a \rceil \equiv 0 \pmod{n}$ *if and only if* $\mathrm{dec}(a) \in \left(1 - \frac{1}{n}, 1\right) \bigcup \{0\}$;

3. $\mathrm{round}(n \cdot a) \equiv 0 \pmod{n}$ *if and only if* $\mathrm{dec}(a) \in \left[0, \frac{1}{2n}\right) \bigcup \left[1 - \frac{1}{2n}, 1\right)$.

This corollary can be derived directly from Lemma 3.1.

Subsequently, we introduce Lemma 3.2, which gives some useful results about the highest $n-i$ bits and the lowest i bits of x/p, where $x, p \in S_n$.

Lemma 3.2: *$\forall p \in D_i = S_i - \{0\}(1 \le i \le n), x \in S_n$. Assume $p = N_p/2^i, x = N_x/2^n$, where N_p, N_x are integers satisfying $1 \le N_p \le 2^i - 1$ and $0 \le N_x \le 2^n - 1$. we have the following three results (where $G_0(\cdot)$ denotes the corresponding ATF of $G_n(\cdot)$, $G_0(\cdot)$ will hold the same meaning in the following contents):*

$$1. \quad G_n(x/p) \in S_{n-i} \Leftrightarrow N_x \equiv 0 \pmod{N_p}, \tag{3.3}$$

$$2. \quad \text{floor}_{n-i}(G_n(x/p)) = \frac{\lfloor N_x/N_p \rfloor}{2^{n-i}}, \tag{3.4}$$

$$3. \quad G_n(x/p) \bmod \frac{1}{2^{n-i}} = \frac{G_0\left(2^i \cdot \frac{N_x \bmod N_p}{N_p}\right)}{2^n}. \tag{3.5}$$

Proof: Because $x/p = \frac{N_x/2^n}{N_p/2^i} = \frac{N_x/N_p}{2^{n-i}} = \frac{\lfloor N_x/N_p \rfloor + (N_x \bmod N_p)/N_p}{2^{n-i}}$, we have $G_n(x/p) = \frac{G_0(2^i \cdot \lfloor N_x/N_p \rfloor + 2^i \cdot (N_x \bmod N_p)/N_p)}{2^n}$. From *ATF Property 1*, we can rewrite $G_n(x/p)$ as follows

$$G_n(x/p) = \frac{\lfloor N_x/N_p \rfloor}{2^{n-i}} + \frac{G_0(2^i \cdot (N_x \bmod N_p)/N_p)}{2^n}. \tag{3.6}$$

Let us discuss the above equation under the following two conditions:

a) When $N_x \bmod N_p = 0$: $G_n(x/p) = \frac{\lfloor N_x/N_p \rfloor}{2^{n-i}} + 0 \in S_{n-i}$;

b) When $N_x \bmod N_p = k \neq 0$: Obviously $1 \le k \le N_p - 1$. Considering $p < 1$, we have $2^i/N_p > 1$, then $1 < 2^i \cdot (N_x \bmod N_p)/N_p < 2^i - 1$. Thus, from *ATF Property 2*, $1 \le G_0(2^i \cdot (N_x \bmod N_p)/N_p) \le 2^i - 1$. Therefore,

$$\frac{\lfloor N_x/N_p \rfloor}{2^{n-i}} + \frac{1}{2^n} \le G_n(x,p) \le \frac{\lfloor N_x/N_p \rfloor}{2^{n-i}} + \frac{2^i - 1}{2^n} \Rightarrow G_n(x,p) \notin S_{n-i}. \tag{3.7}$$

From a) and b), we can deduce $G_n(x/p) \in S_{n-i} \Leftrightarrow N_x \equiv 0 \pmod{N_p}$.

At the same time, when $N_x \bmod N_p = 0$, $\text{floor}_{n-i}(G_n(x/p)) = \frac{\lfloor N_x/N_p \rfloor}{2^{n-i}}$; when $N_x \bmod N_p = k \neq 0$, $\text{floor}_{n-i}(G_n(x/p)) \ge \frac{\left\lfloor \lfloor N_x/N_p \rfloor + 1/2^i \right\rfloor}{2^{n-i}} = \frac{\lfloor N_x/N_p \rfloor}{2^{n-i}}$ and $\text{floor}_{n-i}(G_n(x/p)) \le \frac{\left\lfloor \lfloor N_x/N_p \rfloor + (2^i - 1)/2^i \right\rfloor}{2^{n-i}} = \frac{\lfloor N_x/N_p \rfloor}{2^{n-i}}$, so finally we can get $\text{floor}_{n-i}(G_n(x/p)) = \frac{\lfloor N_x/N_p \rfloor}{2^{n-i}}$.

From the above result and (3.6), the following result is true:

$$G_n(x/p) \bmod \frac{1}{2^{n-i}} = \frac{G_0(2^i \cdot (N_x \bmod N_p)/N_p)}{2^n} .$$

The proof is complete. ■

The following Lemma 3.3 and Corollary 3.2 are also about the digital division x/p realized in S_n. Together with Lemma 3.2, they reflect some essential properties of the digital division x/p in S_n, and play important roles in the following proofs about statistical properties of digital 1D PWLCM-s.

Lemma 3.3: *Assume n is an odd integer, random integer variable K distributes uniformly in $\mathbb{Z}_n = \{0, \cdots, n-1\}$, the following fact is true: $K' = f(K) = (2^i \cdot K) \bmod n$ distributes uniformly in $\mathbb{Z}_n$, i.e., $\forall k \in \{0, \cdots, n-1\}, P\{K' = k\} = 1/n$.*

Proof: As we know, $(\mathbb{Z}_n, +)$ is a finite cyclic group of degree n, and a is its generator if and only if $\gcd(a, n) = 1$, where "+" is defined as "$(a + b) \bmod n$" (see Theorem 2 on page 60 of [211]). Therefore, $a = 2^i \bmod n$ is one generator of $\mathbb{Z}_n$ since $\gcd(a, n) = \gcd(2^i, n) = 1$. Consider $K' = (2^i \cdot K) \bmod n = (a \cdot K) \bmod n$, we can see $f : \mathbb{Z}_n \to \mathbb{Z}_n$ is a bijection. Then we will immediately deduce: $K' = f(K)$ distributes uniformly in $\mathbb{Z}_n$ because K distributes uniformly in $\mathbb{Z}_n$. That is to say, $\forall k \in \{0, \cdots, n-1\}, P\{K' = k\} = 1/n$. The proof is complete. ■

Corollary 3.2: *Assume n is an odd integer, random integer variable K distributes uniformly in $\mathbb{Z}_n = \{0, \cdots, n-1\}$. Then $\mathrm{dec}(2^i \cdot K/n)$ distributes uniformly in $S = \{x | x = k/n, k \in \mathbb{Z}_n\}$.*

This corollary is a straightforward result of Lemma 3.3.

§3.3 Dynamical Indicators of Digital PWLCM-s and Their Exact Calculations

Based on definitions given in §3.2.2, now let us see how to depict a digital normalized 1D PWLCM $F(x) : I \to I$ realized in finite precision n, where $I = [0, 1]$. Consider $1 \notin S_n$ and $1 \equiv 0 \pmod 1$, to facilitate the following description and proofs, we redefine the normalized 1D PWLCM on $[0, 1)$ instead of $[0, 1]$ as follows:

$$F_{[0,1)}(x) = F(x) \bmod 1 = \begin{cases} F(x), & 0 \le F(x) < 1 \\ 0, & F(x) = 1 \end{cases} . \tag{3.8}$$

Such a redefinition will not make nontrivial influence on theoretical results obtained in this chapter, the reason will be explained below. Use $\mathcal{F}_n(x)$ to denote $F_{[0,1)}$ realized in finite precision n, we have $\mathcal{F}_n = G_n \circ F_{[0,1)} : S_n \to S_n$, where $G_n(\cdot)$ is a DATF, i.e., $\text{floor}_n(\cdot)$, $\text{ceil}_n(\cdot)$ or $\text{round}_n(\cdot)$.

§3.3.1 Dynamical Indicators

Now let us give formal definitions of the proposed dynamical indicators. $\forall x = 0.b_n b_{n-1} \cdots b_2 b_1 \in S_n$, define $P_j(x)$ is the probability of the least j bits $b_j \cdots b_1$ are all zeros, or we have an equivalent definition $P_j(x) = P\{x \in S_{n-j}\}$. All following discussions in the present chapter are focused on the following n dynamical indicators:

$$\forall j = 1 \sim n, P_j(\mathcal{F}_n(x)) = P\{\mathcal{F}_n(x) \in S_{n-j}\} \tag{3.9}$$

where $\mathcal{F}_n = G_n \circ F_{[0,1)} : S_n \to S_n$ is a digital normalized 1D PWLCM and x is a discrete variable uniformly distributed in S_n.

If x distributes uniformly in S_n, $P_j(x) = 2^{-j}$. Accordingly, $P_j(\mathcal{F}_n(x)) = 2^{-j}$ if $\mathcal{F}_n(x)$ distributes uniformly in S_n. However, in §3.2.1, we have mentioned that $\mathcal{F}_n(x)$ does not satisfy uniform distribution because of dynamical degradation induced by spatial discretization. That is to say, there exists **at least one** j, which satisfies $P_j(\mathcal{F}_n(x)) \neq 2^{-j}$. Then can we theoretically deduce the exact values of $P_j(\mathcal{F}_n(x))(1 \leq j \leq n)$ to measure such degradation? In this chapter we will give an affirmative answer. The answer reveals some essential and important properties of the discrete iterations of digital 1D PWLCM-s, and may touch the hard kernel of digital arithmetic. Since it is possible to exactly calculate values of $P_j(\mathcal{F}_n(x))(1 \leq j \leq n)$, and the fact at least one $P_j(\mathcal{F}_n(x)) \neq 2^{-j}$, it will be possible $P_1(\mathcal{F}_n(x)) \sim P_n(\mathcal{F}_n(x))$ can reflect non-uniformity degree of $\mathcal{F}_n(x)$ with discrete uniform input x. It is why we call the n probabilities dynamical indicators of digital 1D PWLCM-s.

With the definition of studied dynamical indicators, we can explain why the redefinition (3.8) will not influence the results about $P_j(\mathcal{F}_n(x))$. Although $1 \notin S_n$, we can express 1 as $1.\overbrace{0\cdots0}^{n}$. Comparing 1 with $0 = 0.\overbrace{0\cdots0}^{n}$, we can see 0 and 1 make the same contribution to $P_j(\mathcal{F}_n(x))(1 \leq j \leq n)$. Therefore, the redefinition (3.8) will not change the value of each $P_j(\mathcal{F}_n(x))$.

To simplify description and proofs, in the following contents we will use P_j to denote $P_j(\mathcal{F}_n(x))$. The following contents in this section is divided into four parts: in §3.3.2 we study dynamical indicators $P_j(1 \leq j \leq n)$ on a single linear segment, then dynamical indicators of general digital 1D PWLCM-s are investigated in §3.3.3, the 1D PWLCM (2.1) and (2.3) are given as examples to show

mathematical meanings of the dynamical indicators in §3.3.4, the last subsection discusses dynamical indicators of $\mathcal{F}_n^k(x)$.

§3.3.2 $P_j(1 \leq j \leq n)$ on a Single Linear Segment

Essentially, dynamics of a digital 1D PWLCM is combination of dynamics of its all linear segments. In this subsection, we will study how to calculate $P_j(1 \leq j \leq n)$ on a single linear segment, where $\mathcal{F}_n(x) = G_n(x/p), x \in C = [0, p) \cap S_n$. Because each linear segment of a 1D PWLCM can be reduced into the form x/p by linear transformation, dynamical indicators of this PWLCM can be calculated by combing $P_j(1 \leq j \leq n)$ on each linear segment. Here, please note that results given in this subsection are also available for any 1D PWLM.

Lemma 3.4: *Assume a discrete random variable x distributes uniformly in the discrete set $C = [0, p) \cap S_n$ and $p = N_p/2^i \in D_i = S_i - \{0\}$, where N_p is an integer in $\{1, \cdots, 2^i - 1\}$. For a digital linear function $\mathcal{F}_n(x) = G_n(x/p)$, $\text{floor}_{n-i}(\mathcal{F}_n(x))$ distributes uniformly in S_{n-i}, i.e., $\forall k \in \{0, \cdots, 2^{n-i} - 1\}$, $P\left\{\text{floor}_{n-i}(\mathcal{F}_n(x)) = k/2^{n-i}\right\} = 1/2^{n-i}$.*

Proof: Assume $x = N_x/2^n$, from $x \in [0, p) \cap S_n$ and $p = N_p/2^i$, we can deduce $0 \leq N_x \leq 2^{n-i} \cdot N_p - 1$. Because x distributes uniformly in C, N_x will distribute uniformly in the integer set $\{0, \cdots, 2^{n-i} \cdot N_p - 1\}$.

Consider $\mathcal{F}_n(x) = G_n(x/p)$, from Eq. (3.4) of Lemma 3.2, we have $\text{floor}_{n-i}(\mathcal{F}_n(x)) = \lfloor N_x/N_p \rfloor / 2^{n-i}$. Since N_x distributes uniformly in $\{0, \cdots, 2^{n-i} \cdot N_p - 1\}$, $\lfloor N_x/N_p \rfloor$ will also distribute uniformly in $\{0, \cdots, 2^{n-i} - 1\}$, i.e., $\text{floor}_{n-i}(\mathcal{F}_n(x))$ distributes uniformly in S_{n-i}. The proof is complete. ■

Lemma 3.5: *Assume a discrete random variable x distributes uniformly in the discrete set $C = [0, p) \cap S_n$ and $p = N_p/2^i \in D_i = S_i - \{0\}$, where N_p is an integer in $\{1, \cdots, 2^i - 1\}$. For a digital linear function $\mathcal{F}_n(x) = G_n(x/p)$, we have: $i \leq j \leq n$, $P_j = 1 \left/ \left(N_p \cdot 2^{j-i}\right)\right.$.*

Proof: Similar to the proof of Lemma 3.4, assume $x = N_x/2^n$, we can deduce N_x distributes uniformly in the integer set $\{0, \cdots, 2^{n-i} \cdot N_p - 1\}$. Then let us respectively consider the following two conditions:

a) $j = i$: Because $\mathcal{F}_n(x) = G_n(x/p)$, from Eq. (3.3) of Lemma 3.2, we know $\mathcal{F}_n(x) \in S_{n-i}$ if and only if $N_x \equiv 0 \pmod{N_p}$. Consider there are 2^{n-i} integers satisfying $N_x \equiv 0 \pmod{N_p}$ and N_x distributes uniformly in $\{0, \cdots, 2^{n-i} \cdot N_p - 1\}$, the probability of $\mathcal{F}_n(x) \in S_{n-i}$ will be $\dfrac{2^{n-i}}{2^{n-i} \cdot N_p} = \dfrac{1}{N_p}$. That is to say, $P_i = \dfrac{1}{N_p} = \dfrac{1}{N_p \cdot 2^{i-i}}$.

b) $i+1 \le j \le n$: Assume $\mathcal{F}_n(x) = 0.b_n b_{n-1}\cdots b_2 b_1$, $P_j = P\left\{\mathcal{F}_n(x) \in S_{n-i} \wedge b_j \cdots b_{i+1} = \overbrace{0\cdots 0}^{j-i}\right\}$. Recall the proof of Lemma 3.4, we can know the event $\mathcal{F}_n(x) \in S_{n-i}$ is independent of the event $b_j \cdots b_{i+1} = \overbrace{0\cdots 0}^{j-i}$, so $P_j = P\{\mathcal{F}_n(x) \in S_{n-i}\} \cdot P\left\{b_j \cdots b_{i+1} = \overbrace{0\cdots 0}^{j-i}\right\}$. From Lemma 3.4, the highest $n-i$ bits of $F_n(x,p)$ distributes uniformly in $\{0,\cdots,2^{n-i}-1\}$, thus $P\left\{b_j \cdots b_{i+1} = \overbrace{0\cdots 0}^{j-i}\right\} = \frac{1}{2^{j-i}}$. Finally, we have $P_j = P_i \cdot \frac{1}{2^{j-i}} = \frac{1}{N_p \cdot 2^{j-i}}$.

As a whole, $i \le j \le n$, $P_j = \frac{1}{N_p \cdot 2^{j-i}}$. ■

Lemma 3.6: *Assume a discrete random variable x distributes uniformly in the discrete set $C = [0,p) \cap S_n$ and $p = N_p/2^i \in V_i (1 \le i \le n)$ (please note **not D_i** in the above two lemmas), where N_p is an **odd** integer in $\{1,\cdots,2^i-1\}$. For a digital linear function $\mathcal{F}_n(x) = G_n(x/p)$, we have:*

$$1 \le j \le i-1, P_j = \begin{cases} \dfrac{\lfloor N_p/2^j \rfloor + 1}{N_p}, & G_n(\cdot) = \text{floor}_n(\cdot) \text{ or } \text{ceil}_n(\cdot) \\ \dfrac{2\cdot\lfloor N_p/2^{j+1} \rfloor + 1}{N_p}, & G_n(\cdot) = \text{round}_n(\cdot) \end{cases}. \quad (3.10)$$

Proof: Similar to the proof of Lemma 3.4, assume $x = N_x/2^n$, N_x will distribute uniformly in the integer set $\{0,\cdots,2^{n-i}\cdot N_p - 1\}$.

Because $\mathcal{F}_n(x) = G_n(x/p)$, from Eq. (3.5) of Lemma 3.2, we know the least i bits of $\mathcal{F}_n(x)$ are determined by $G_0\left(2^i \cdot \frac{N_x \bmod N_p}{N_p}\right)$. Then we can deduce $\mathcal{F}_n(x) \in S_{n-j} \Leftrightarrow G_0\left(2^i \cdot \frac{N_x \bmod N_p}{N_p}\right) \equiv 0 \pmod{2^j}$. Define $\hat{N}_x = N_x \bmod N_p$, which distributes uniformly in $\{0,\cdots,N_p-1\}$ because of the uniform distribution of N_x. Then define $a = \frac{2^i \cdot \hat{N}_x/N_p}{2^j}$, we can rewrite $G_0\left(2^i \cdot \frac{N_x \bmod N_p}{N_p}\right)$ as

$G_0(2^j \cdot a)$. From Corollary 3.1, we have:

$$\begin{array}{c} G_0(2^j \cdot a) \equiv 0 \pmod{2^j} \\ \Updownarrow \\ \mathrm{dec}(a) \in \begin{cases} \left[0, \frac{1}{2^j}\right), & G_0(\cdot) = \lfloor \cdot \rfloor \\ \left(1 - \frac{1}{2^j}, 1\right) \bigcup \{0\}, & G_0(\cdot) = \lceil \cdot \rceil \\ \left[0, \frac{1}{2^{j+1}}\right) \bigcup \left[1 - \frac{1}{2^{j+1}}, 1\right), & G_0(\cdot) = \mathrm{round}(\cdot) \end{cases} . \end{array} \tag{3.11}$$

From Corollary 3.2 ($p \in \boldsymbol{V_i}$ ensures N_p is an **odd** integer), we know $\mathrm{dec}(a)$ distributes in $\{0, \cdots, N_p - 1\}$ uniformly, i.e. $\forall k = 0 \sim N_p - 1$, $P\left\{\mathrm{dec}(a) = \frac{k}{N_p}\right\} = \frac{1}{N_p}$. That is to say, assume $\hat{N}'_x = \mathrm{dec}(a) \cdot N_p = \frac{2^i \cdot \hat{N}_x}{2^j}$, we have $P\{\hat{N}'_x = k\} = \frac{1}{N_p}$. Based on (3.11), we can deduce:

$$\begin{array}{c} G_0(2^j \cdot a) \equiv 0 \pmod{2^j} \\ \Updownarrow \\ \hat{N}'_x \in \begin{cases} \left[0, \frac{N_p}{2^j}\right), & G_0(\cdot) = \lfloor \cdot \rfloor \\ \left(N_p - \frac{N_p}{2^j}, N_p\right) \bigcup \{0\}, & G_0(\cdot) = \lceil \cdot \rceil \\ \left[0, \frac{N_p}{2^{j+1}}\right) \bigcup \left[N_p - \frac{N_p}{2^{j+1}}, N_p\right), & G_0(\cdot) = \mathrm{round}(\cdot) \end{cases} . \end{array} \tag{3.12}$$

Consider $\hat{N}'_x$ is an integer, we can further get:

$$\begin{array}{c} G_0(2^j \cdot a) \equiv 0 \pmod{2^j} \\ \Updownarrow \\ \hat{N}'_x \in \begin{cases} \left\{0, \cdots, \left\lfloor \frac{N_p}{2^j} \right\rfloor\right\}, & G_0(\cdot) = \lfloor \cdot \rfloor \\ \{0\} \bigcup \left\{N_p - \left\lfloor \frac{N_p}{2^j} \right\rfloor, \cdots, N_p - 1\right\}, & G_0(\cdot) = \lceil \cdot \rceil \\ \left\{0, \cdots, \left\lfloor \frac{N_p}{2^{j+1}} \right\rfloor\right\} \bigcup \left\{N_p - \left\lfloor \frac{N_p}{2^{j+1}} \right\rfloor, \cdots, N_p - 1\right\}, & G_0(\cdot) = \mathrm{round}(\cdot) \end{cases} . \end{array} \tag{3.13}$$

From the uniform distribution of $\hat{N}'_x$ in $\{0, \cdots, N_p - 1\}$, we can easily deduce the value of P_j as follows:

$$\begin{aligned} P_j &= P\left\{\mathcal{F}_n(x) \in S_{n-j}\right\} \\ &= P\left\{G_0(2^j \cdot a) \equiv 0 \pmod{2^j}\right\} \end{aligned}$$

$$= \begin{cases} \dfrac{\lfloor N_p/2^j \rfloor + 1}{N_p}, & G_0(\cdot) = \lfloor \cdot \rfloor \text{ or } \lceil \cdot \rceil \\ \dfrac{2 \cdot \lfloor N_p/2^{j+1} \rfloor + 1}{N_p}, & G_0(\cdot) = \text{round}(\cdot) \end{cases}. \tag{3.14}$$

Apparently, Eq. (3.10) holds. The proof is complete. ■

Theorem 3.1: *Assume a discrete random variable x distributes uniformly in the discrete set $C = [0, p) \cap S_n$ and $p = N_p/2^i \in V_i (1 \le i \le n)$ (N_p is an **odd** integer in $\{1, \cdots, 2^i - 1\}$). For a digital linear function $\mathcal{F}_n(x) = G_n(x/p)$, we have:*

$$P_j = \left\{ \begin{array}{ll} & \dfrac{1}{N_p \cdot 2^{j-i}} \quad , \quad i \le j \le n \\ \dfrac{\lfloor N_p/2^j \rfloor + 1}{N_p} & , \quad G_n(\cdot) = \text{floor}_n(\cdot) \text{ or } \text{ceil}_n(\cdot) \\ \dfrac{2 \cdot \lfloor N_p/2^{j+1} \rfloor + 1}{N_p}, & \quad G_n(\cdot) = \text{round}_n(\cdot) \end{array} \right\}, \quad 1 \le j \le i-1 \tag{3.15}$$

Proof: This theorem can be directly derived from Lemma 3.5 and 3.6. ■

§3.3.3 $P_j(1 \le j \le n)$ of digital 1D PWLCM-s

How to Calculate values of n Dynamical Indicators?

Based on $P_j(1 \le j \le n)$ of the digital linear function $\mathcal{F}_n(x) = G_n(x/p)$, we can calculate the exact values of $P_j(1 \le j \le n)$ of a digital 1D PWLCM. Given a normalized 1D PWLCM denoted by Eq. (3.1), we can rewrite the linear segment $F_i(x) = a_i x + b_i$ as follows: $F_i(x_i') = x_i'/p_i$, $x_i' \in [0, p_i)$, where $p_i = 1/|a_i|$, $x_i' = \text{sign}(a_i) \cdot (x + b_i/a_i)$. Here $p_i \in (0, 1) \subset [0, 1] = S_\infty$ since $|a_i| > 1$. Together with the redefinition (3.8), we can rewrite the 1D PWLCM as follows:

$$i = 1 \sim m, F(x_i')|C_i' = F_i(x_i') = x_i'/p_i, x_i' \in C_i' = [0, p_i) \tag{3.16}$$

When the 1D PWLCM is realized in finite precision n, F_i is denoted by $\mathcal{F}_n^{(i)}$.

Assume $p_i = N_{p_i}/2^{r_i} \in V_{r_i}$, where r_i is the resolution of p_i. Denote the probability of $P_j|x \in C_i$ as $P_j^{(i)}$, from the total probability rule[212], the j^{th} dynamical indicator P_j of the digital 1D PWLCM will be:

$$P_j = \sum_{i=1}^{m} P_j^{(i)} \cdot \|C_i\| = \sum_{i=1}^{m} P_j^{(i)} \cdot |p_i| = \sum_{i=1}^{m} P_j^{(i)} \cdot \frac{N_{p_i}}{2^{r_i}}. \tag{3.17}$$

Assume $\mathcal{P}_j^{(i)} = P_j^{(i)} \cdot \|C_i\|$, we have $P_j = \sum_{i=1}^{m} \mathcal{P}_j^{(i)}$. From Theorem 3.1, we can easily get:

$$\mathcal{P}_j^{(i)} = \left\{ \begin{array}{cc} 1/2^j & , \quad r_i \le j \le n \\ \left. \begin{array}{cc} \frac{\lfloor N_{p_i}/2^j \rfloor + 1}{2^{r_i}} & , \quad G_n(\cdot) = \text{floor}_n(\cdot) \text{ or } \text{ceil}_n(\cdot) \\ \frac{2 \cdot \lfloor N_{p_i}/2^{j+1} \rfloor + 1}{2^{r_i}}, & G_n(\cdot) = \text{round}_n(\cdot) \end{array} \right\}, & 1 \le j \le r_i - 1 \end{array} \right. . \tag{3.18}$$

Thus, we can get the values of P_j when $\max_{i=1}^{m}(r_i) \le j \le n$:

$$P_j = \frac{m}{2^j}, \tag{3.19}$$

and the values of P_j when $1 \le j \le \min_{i=1}^{m}(r_i) - 1$:

$$P_j = \left\{ \begin{array}{cc} \sum_{i=1}^{m} \frac{\lfloor N_{p_i}/2^j \rfloor + 1}{2^{r_i}} & , \quad G_n(\cdot) = \text{floor}_n(\cdot) \text{ or } \text{ceil}_n(\cdot) \\ \sum_{i=1}^{m} \frac{2 \cdot \lfloor N_{p_i}/2^{j+1} \rfloor + 1}{2^{r_i}}, & G_n(\cdot) = \text{round}_n(\cdot) \end{array} \right. . \tag{3.20}$$

When $\min_{i=1}^{m}(r_i) \le j \le \max_{i=1}^{m}(r_i) - 1$, we can calculate the exact value of each $\mathcal{P}_j^{(i)}$ by Eq. (3.18) to get P_j.

In the following, let us analyze how P_j reflects the dynamical degradation of digital 1D PWLCM-s and how P_j changes as j changes. Here, we use $\overline{\boldsymbol{P}}_{\boldsymbol{j}}$ to denote the *balanced* dynamical indicator 2^{-j} when $\mathcal{F}_n(x)$ uniformly distributes in S_n.

How Do Dynamical Indicators Change as j Changes?

When $\max_{i=1}^{m}(r_i) \le j \le n$, P_j is m times of $\overline{\boldsymbol{P}}_{\boldsymbol{j}}$, where m is the number of the linear segments of $\mathcal{F}_n(x)$. Since $m \ge 2$, we can see the above fact reflects the essential non-uniformity of $\mathcal{F}_n(x)$ in S_n. Now, P_j is not only independent of resolutions of $p_1, \cdots, p_m$, but independent of their exact values and the selection of DATF.

When $1 \le j \le \min_{i=1}^{m}(r_i) - 1$, the values of P_j are dependent on the exact values of $p_1, \cdots, p_m$ and the selection of DATF. Although we cannot calculate the exact values when $p_1 \sim p_m$ are not known, we can still deduce the upper bound and the lower bound of P_j. Because N_{p_i} is an odd integer, both $N_{p_i}/2^j$ and $N_{p_i}/2^{j+1}$ are not integers, then we have*:

$$\begin{aligned} N_{p_i}/2^j - 1 < \lfloor N_{p_i}/2^j \rfloor < N_{p_i}/2^j, \\ N_{p_i}/2^{j+1} - 1 < \lfloor N_{p_i}/2^{j+1} \rfloor < N_{p_i}/2^{j+1}. \end{aligned} \tag{3.21}$$

*$\forall a \in \mathbb{R} - \mathbb{Z}$, we have $a - 1 < \lfloor a \rfloor < a$, which is the natural result of the definition of the floor function.

Substitute the above inequalities into Eq. (3.20) and consider $\sum_{i=1}^{m}|p_i| = \sum_{i=1}^{m}\|C_i\| = 1 \Rightarrow \sum_{i=1}^{m} N_{p_i}/2^{r_i} = 1$*, we can obtain the following results:

$$\text{When } G_n(\cdot) = \text{floor}_n(\cdot) \text{ or } \text{ceil}_n(\cdot) \quad , \quad \frac{1}{2^j} < P_j < \frac{1}{2^j} + \sum_{i=1}^{m}\frac{1}{2^{r_i}}. \tag{3.22}$$

$$\text{When } G_n(\cdot) = \text{round}_n(\cdot) \quad , \quad \frac{1}{2^j} - \sum_{i=1}^{m}\frac{1}{2^{r_i}} < P_j < \frac{1}{2^j} + \sum_{i=1}^{m}\frac{1}{2^{r_i}}. \tag{3.23}$$

Generally speaking, the greater $r_1, \cdots, r_m$ are, the closer P_j will be to $\overline{\boldsymbol{P}}_{\boldsymbol{j}} = 2^{-j}$, i.e., the smaller $P_j - 2^{-j}$ will be. Here, please note that P_j may be exactly $\overline{\boldsymbol{P}}_{\boldsymbol{j}} = 2^{-j}$ when $G_n(\cdot) = \text{round}_n(\cdot)$, one example is the 1D PWLCM (2.1) and the skew tent map (2.3) (we will prove these results in the next sub-section).

At last let us investigate values of P_j when $\min_{i=1}^{m}(r_i) \le j \le \max_{i=1}^{m}(r_i) - 1$. Apparently, now P_j will be also dependent on $p_1, \cdots, p_m$ and the selection of $G_n(\cdot)$, but such dependence will be weaker compared with P_j when $1 \le j \le \min_{i=1}^{m}(r_i) - 1$. What's more, the smaller j is, the stronger the dependence will be.

Observe the values of P_j for $\max_{i=1}^{m}(r_i) \le j \le n$ and for $1 \le j \le \min_{i=1}^{m}(r_i) - 1$, we can **conceptually** and **intuitively** deduce the following fact: as j goes from n to $\max_{i=1}^{m}(r_i)$, P_j preserves fixed m times of $\overline{\boldsymbol{P}}_{\boldsymbol{j}} = 2^{-j}$; as j goes to 1 from $\max_{i=1}^{m}(r_i)$, P_j tends to have less and less times of $\overline{\boldsymbol{P}}_{\boldsymbol{j}} = 2^{-j}$. Of course, for different digital 1D PWLCM-s, the actual properties may be different, but the above result is right **roughly**.

How Do We Understand Relation between the Indicators and Dynamical Degradation of Digital PWLCM-s?

As a whole, when $G_n(\cdot) = \text{round}_n(\cdot)$, at least $n + 1 - \max_{i=1}^{m}(r_i)$ indicator(s) satisfy $P_j \neq 1/2^j$; and when $G_n(\cdot) = \text{floor}_n(\cdot)$ or $\text{ceil}_n(\cdot)$, all n indicators satisfy $P_j \neq 1/2^j$. Consider $P_j = m/2^j$ for $\max_{i=1}^{m}(r_i) \le j \le n$, the dynamical degradation of a digital 1D PWLCM can be qualitatively measured by the number of the linear segments: m. That is to say, the lager m is, the more serious the dynamical degradation of a digital PWLCM will be.

Another function of the dynamical indicators is to distinguish different dynamical degradation of different control parameters. For a given digital 1D PWLCM, let us find the relation between the dynamical degradation and the resolution r_i of the control parameter p_i. For the set of m control parameters $\boldsymbol{p} = \{p_1, p_2, \ldots, p_m\}$, define $\widetilde{P} = \frac{1}{n} \cdot \sum_{j=1}^{n} \frac{P_j}{\overline{\boldsymbol{P}}_{\boldsymbol{j}}}$ as the *average degradation factor*

*Please note that this result only holds for PWLCM-s with onto property, and is not true for general PWLM-s.

of $\boldsymbol{p}$, which can quantitatively reflect the dynamical degradation of digital 1D PWLCM-s with the control parameters set $\{p_1, p_2, \ldots, p_m\}$. Apparently, the larger $\widetilde{P}$ is, the more serious dynamical degradation will be. For two digital 1D PWLCM-s $\mathcal{F}_n(x)$ and $\mathcal{F}'_n(x)$ with different control parameters sets $\boldsymbol{p}$ and $\boldsymbol{p'}$, if $\widetilde{P} > \widetilde{P}'$, we say $\boldsymbol{p}$ is *weaker* than $\boldsymbol{p'}$ (or $\boldsymbol{p'}$ is *stronger* than $\boldsymbol{p}$), which is denoted by $\boldsymbol{p} \prec \boldsymbol{p'}$ (or $\boldsymbol{p'} \succ \boldsymbol{p}$). If $P_j > P'_j$, we say $\boldsymbol{p}$ is *weaker at resolution* j than $\boldsymbol{p'}$ (or $\boldsymbol{p'}$ is *stronger at resolution* j than $\boldsymbol{p}$), which is denoted by $\boldsymbol{p} \prec_j \boldsymbol{p'}$ (or $\boldsymbol{p'} \succ_j \boldsymbol{p}$). For a single control parameter $p_i (1 \leq i \leq m)$, the relation of $\prec$ and $\prec_j$ can also similarly be defined under such an assumption that all other control parameters are uniformly distributed in the parameter space. From the above discussion, we can see the following fact: the smaller the resolution r_i is, the weaker the control parameter p_i will be.

From the above discussion, since $P_j \neq 2^{-j}$ means non-uniformity of chaotic output, the proposed dynamical indicators can be considered as statistical measures of pseudo-ergodicity of digital chaotic PWLCM-s, and also an evidence of measurable discrepancy of digital invariant measure from its continuous counterpart. In the following subsection, from two concrete examples, we will show that an interesting fact on digital 1D PWLCM-s: the smaller resolutions of all linear slopes are, the larger $|P_j - \overline{\boldsymbol{P}}_{\boldsymbol{j}}|$ will be. What on earth does small resolution mean? Let us rewrite a linear slop p with resolution i as $p = \dfrac{N_p}{2^i} = 2^{n-i} \cdot \dfrac{N_p}{2^n}$, we can see small resolution i means a large factor 2^{n-i}. When we do digital divisions x/p with n-bit fixed-point arithmetic, assume $x = N_x/2^n$, the division can be expressed as $x/p = 2^{n-i} \cdot \dfrac{N_x}{N_p}$, where 2^{n-i} means left shifting operation and apparently will promote the value of each dynamical indicator. Essentially speaking, these indicators reflect the collapse of digital (fixed-point) divisions on each linear segment and accumulation of such collapse of multiple linear segments. As a natural result, such collapse of digital arithmetic further causes collapse of dynamics of digital PWLCM-s.

Especially, if the explicit equation of a digital 1D PWLCM is known, more delicate results may be obtained. In the next subsection, we will give the exact values of $P_j (1 \leq j \leq n)$ of the 1D PWLCM (2.1) and the skew tent map (2.3)*. For the two 1D PWLCM-s, all n values of $P_j (1 \leq j \leq n)$ are uniquely determined by the resolution of the control parameter p, and independent of its exact value. Because only one control parameter is concerned, some interesting and meaningful facts about P_j of digital 1D PWLCM-s can be shown clearly.

*Although we have proved corresponding results on the two classes of digital PWLCM-s in [109], the proofs given in §3.3.4 are based on Eq. (3.19), (3.20) and somewhat different from the ones in [109].

§3.3.4 Two Concrete Examples

To calculate the exact values of $P_j(1 \le j \le \min_{i=1}^{m}(r_i) - 1)$ of the digital 1D PWLCM (2.1) and (2.3), we should firstly introduce a new lemma.

Lemma 3.7: *$\forall j, N, N' \in \mathbb{Z}^+$, N, N' are odd integers and $2^j | (N + N')$, we have $\lfloor N/2^j \rfloor + \lfloor N'/2^j \rfloor = (N + N')/2^j - 1$.*

Proof: Because $a = \lfloor a \rfloor + \mathrm{dec}(a)$, $\lfloor N/2^j \rfloor + \lfloor N'/2^j \rfloor = \left(N/2^j - \mathrm{dec}(N/2^j)\right) + \left(N'/2^j - \mathrm{dec}(N'/2^j)\right)$. Assume $N = n_1 \cdot 2^j + n_2, N' = n_1' \cdot 2^j + n_2'$ and $N + N' = 2^k (k \ge j)$, we have $\mathrm{dec}(N/2^j) = (N \mod n)/2^j = n_2/2^j, \mathrm{dec}(N'/2^j) = (N' \mod n)/2^j = n_2'/2^j$. Since N, N' are odd integers, we can get $n_2 > 0, n_2' > 0$. From $2^j | (N + N')$, it is obvious that $n_2 + n_2' = 2^j \Rightarrow \mathrm{dec}(N/2^j) + \mathrm{dec}(N'/2^j) = 1$, thus $\lfloor N/2^j \rfloor + \lfloor N'/2^j \rfloor = (N + N')/2^j - 1$. The proof is complete. ■

$P_j(1 \le j \le n)$ of the Digital 1D PWLCM (2.1)

Note: Consider $0 < p < 1/2$, the resolution of p will be in $\{2, \ldots, n\}$.

Theorem 3.2: *Assume a discrete random variable x distributes uniformly in S_n. $\forall p \in V_i (2 \le i \le n)$, the following results are true for the digital 1D PWLCM (2.1):*

1. *When $G_n(\cdot) = \mathrm{round}_n(\cdot)$, $P_j = \begin{cases} 4/2^j, & i \le j \le n \\ 4/2^i, & j = i - 1 \\ 1/2^j, & 1 \le j \le i - 2 \end{cases}$;*

   When $G_n(\cdot) = \mathrm{floor}_n(\cdot)$ or $\mathrm{ceil}_n(\cdot)$, $P_j = \begin{cases} 4/2^j, & i \le j \le n \\ 1/2^j + 2/2^i, & 1 \le j \le i - 1 \end{cases}$;

2. *$\forall k \in \{0, \cdots, 2^{n-i} - 1\}$, $P\left\{\mathrm{floor}_{n-i}(F_n(x, p)) = k/2^{n-i}\right\} = 1/2^{n-i}$.*

Proof: For the 1D PWLCM (2.1), $m = 4$. The slopes of the four linear segments are: $p_1 = p_4 = p$ and $p_2 = p_3 = 1/2 - p$. Since $p \in V_i$, $r_1 = r_2 = r_3 = r_4 = i$ and $\max_{i=1}^{4}(r_i) = \min_{i=1}^{4}(r_i) = i$.

When $i \le j \le n$, from Eq. (3.19), we can easily get

$$P_j = 4/2^j. \tag{3.24}$$

When $1 \le j \le i - 1$, we separately consider two different conditions: $G_n(\cdot) = \mathrm{floor}_n(\cdot)$ or $\mathrm{ceil}_n(\cdot)$, and $G_n(\cdot) = \mathrm{round}_n(\cdot)$.

i) $G_n(\cdot) = \mathrm{floor}_n(\cdot)$ or $\mathrm{ceil}_n(\cdot)$: From Eq. (3.20), we have

$$\begin{aligned} P_j &= \sum_{i=1}^{4} \frac{\lfloor N_{p_i}/2^j \rfloor + 1}{2^i} \\ &= 2 \cdot \sum_{i=1}^{2} \frac{\lfloor N_{p_i}/2^j \rfloor + 1}{2^i} \\ &= 2 \cdot \frac{\lfloor N_{p_1}/2^j \rfloor + \lfloor N_{p_2}/2^j \rfloor + 2}{2^i}. \end{aligned} \tag{3.25}$$

Because $p_1 + p_2 = 1/2 \Rightarrow N_{p_1} + N_{p_2} = 2^{i-1} \Rightarrow 2^j | (N_{p_1} + N_{p_2})$, from Lemma 3.7, we can deduce:

$$\begin{aligned} P_j &= 2 \cdot \frac{(N_{p_1} + N_{p_2})/2^j - 1 + 2}{2^i} \\ &= 2 \cdot \frac{2^{i-1-j} + 1}{2^i} = \frac{1}{2^j} + \frac{2}{2^i}. \end{aligned} \tag{3.26}$$

ii) $G_n(\cdot) = \text{round}_n(\cdot)$: From Eq. (3.20), we have

$$\begin{aligned} P_j &= \sum_{i=1}^{4} \frac{2 \cdot \lfloor N_{p_i}/2^{j+1} \rfloor + 1}{2^i} \\ &= 2 \cdot \sum_{i=1}^{2} \frac{2 \cdot \lfloor N_{p_i}/2^{j+1} \rfloor + 1}{2^i} \\ &= 2 \cdot \frac{2(\lfloor N_{p_1}/2^{j+1} \rfloor + \lfloor N_{p_2}/2^{j+1} \rfloor) + 2}{2^i} \\ &= 4 \cdot \frac{\lfloor N_{p_1}/2^{j+1} \rfloor + \lfloor N_{p_2}/2^{j+1} \rfloor + 1}{2^i}. \end{aligned} \tag{3.27}$$

When $j < i - 1$, $N_{p_1} + N_{p_2} = 2^{i-1} \Rightarrow 2^{j+1} | (N_p + N'_p)$, from Lemma 3.7, we can get:

$$\begin{aligned} P_j &= 4 \cdot \frac{(N_{p_1} + N_{p_2})/2^{j+1} - 1 + 1}{2^i} \\ &= 4 \cdot \frac{2^{i-j-2}}{2^i} = \frac{1}{2^j}. \end{aligned} \tag{3.28}$$

When $j = i - 1$, $N_{p_1} + N_{p_2} = 2^{i-1} \Rightarrow 2^{j+1} \nmid (N_{p_1} + N_{p_2})(j + 1 = i > i - 1)$, Lemma 3.7 cannot be used, but we can directly calculate the probability P_j as follows: $N_{p_1} < 2^i, N_{p_2} < 2^i$, so $N_{p_1}/2^{j+1} < 1 \Rightarrow \lfloor N_{p_1}/2^{j+1} \rfloor = 0, N_{p_2}/2^{j+1} < 1 \Rightarrow \lfloor N_{p_2}/2^{j+1} \rfloor = 0$, then we have

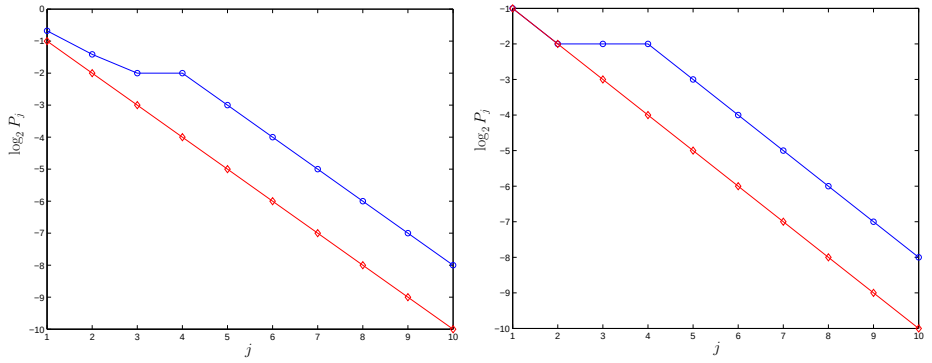
$$P_j = 4 \cdot \frac{0 + 0 + 1}{2^i} = \frac{4}{2^i}. \tag{3.29}$$

From (3.24) – (3.29), we can know the first result is right. In addition, the second result can be directly derived from Lemma 3.4. The proof is complete. ■

Theorem 3.2 tells us the following fact: If x distributes uniformly in S_n, the digital 1D PWLCM (2.1) does not distribute uniformly in S_n; but $\mathcal{F}_n(x)$'s highest $n - i$ bits does in S_{n-i}, $\forall p \in S_i$. To understand what this theorem really means, see Figure 3.1 for visual view.

From Theorem 3.2, we can also derive the rigorous relation between the dynamical degradation and the resolution i of the control parameter p : the smaller the resolution i is, the weaker p will be (see Figure 3.2). For arithmetic explanation of this fact, please see discussion in the last subsection.

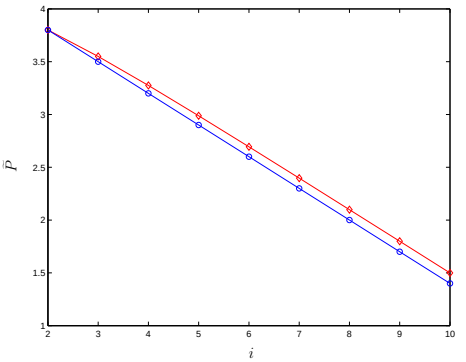
Corollary 3.3: *For the digital 1D PWLCM (2.1), given two different control parameters $p \in V_i, p' \in V_{i'}$, where $i, i' = 2 \sim n$. We have: $i < i' \Leftrightarrow p \prec p'$.*



a) $G_n(\cdot) = \text{floor}_n(\cdot)$ or $\text{ceil}_n(\cdot)$ b) $G_n(\cdot) = \text{round}_n(\cdot)$

$\circ$: $\log_2 P_j$, $\diamond$: $\log_2 \overline{\boldsymbol{P}}_{\boldsymbol{j}} = -j$

Figure 3.1: $\log_2 P_j (1 \leq j \leq n)$ when $p = 3/16 \in V_4 \subset S_4$, where the finite precision $n = 10$



$\circ$: $G_n(\cdot) = \text{round}_n(\cdot)$, $\diamond$: $G_n(\cdot) = \text{floor}_n(\cdot)$ or $\text{ceil}_n(\cdot)$

Figure 3.2: $\widetilde{P}$ versus resolution i, where $n = 10$

Proof: We consider the following two conditions:

a) When $G_n(\cdot) = \text{round}_n(\cdot)$,

$$\frac{P_j}{\overline{\boldsymbol{P}}_{\boldsymbol{j}}} = \frac{P_j}{2^{-j}} = \begin{cases} 4, & i \leq j \leq n \\ 2, & j = i-1 \\ 1, & 1 \leq j \leq i-2 \end{cases} . \tag{3.30}$$

Then we can deduce the value of $\widetilde{P}$:

$$\begin{aligned} \widetilde{P} &= \frac{1}{n} \cdot \sum_{j=1}^{n} \frac{P_j}{\overline{\boldsymbol{P}}_{\boldsymbol{j}}} \\ &= \frac{1}{n} \cdot (4 \cdot (n-i+1) + 2 + 1 \cdot (i-2)) \end{aligned}$$

$$= \quad 4\left(1+\frac{1}{n}\right)-\frac{3i}{n}. \tag{3.31}$$

b) When $G_n(\cdot)=\text{floor}_n(\cdot)$ or $\text{ceil}_n(\cdot)$,

$$\frac{P_j}{\overline{\boldsymbol{P}_j}}=\frac{P_j}{2^{-j}}=\begin{cases} 4 & , \quad i\le j\le n \\ 1+2^{j-(i-1)}, & 1\le j\le i-1 \end{cases}. \tag{3.32}$$

Then we can deduce the value of $\widetilde{P}$:

$$\begin{aligned} \widetilde{P} &= \frac{1}{n}\cdot\sum_{j=1}^{n}\frac{P_j}{\overline{\boldsymbol{P}_j}} \\ &= \frac{1}{n}\cdot\left(4\cdot(n-i+1)+\sum_{j=1}^{i-1}\left(1+\frac{2^j}{2^{i-1}}\right)\right) \\ &= \frac{1}{n}\cdot\left(4\cdot(n-i+1)+(i-1)+2\left(1-\frac{1}{2^{i-1}}\right)\right) \\ &= \left(4+\frac{5}{n}\right)-\frac{1}{n}\cdot\left(3i+\frac{4}{2^i}\right). \end{aligned} \tag{3.33}$$

We can see $\widetilde{P}$ is a descending function of i for any DATF $G_n(\cdot)$. That is to say, $i<i' \Leftrightarrow \widetilde{P}>\widetilde{P}' \Leftrightarrow p\prec p'$. The proof is complete. ■

Remark 3.1: *Note there is an* ***absolutely weak*** *control parameter $p=1/4\in V_2$, which satisfies $P_1=P_2=4/2^2=1$. That is to say, the least 2 bits of $\mathcal{F}_n(x)$ will always be zeros when $p=1/4$. In addition, $\forall x_0\in V_i(2\le i\le n)$, after $\lceil i/2\rceil$ iterations, the chaotic orbit will converge at zero: $\forall k\ge\lceil i/2\rceil, \mathcal{F}_n^k(x_0)=0$. Such a special 1D PWLCM is the four-linear-segment version of the tent map $F(x)=1-2|x-1/2|$, whose digital dynamical properties have been discussed as an extreme example of dynamical degradation of digital chaotic system in §2.5.*

Theorem 3.3: *Assume a discrete random variable x distributes uniformly in S_n. $\forall p\in(0,1/2)\cap S_n$, the following results are true for the digital 1D PWLCM (2.1):*

1. $\forall p\in D_{i,1}=S_i-S_1=\bigcup_{k=2}^{i}V_i$, $P_i=4/2^i$;
2. $\forall p\in V_{i+1}$, $P_i=2/2^i$;
3. $\forall p\in V_j(j\ge i+2)$, $P_i=\begin{cases} 1/2^i & , G_n(\cdot)=\text{round}_n(\cdot) \\ 1/2^i+2/2^j, & G_n(\cdot)=\text{floor}_n(\cdot)\ \textit{or}\ \text{ceil}_n(\cdot) \end{cases}$.

Proof: This theorem is an equivalent form of the first result of Theorem 3.2. ■

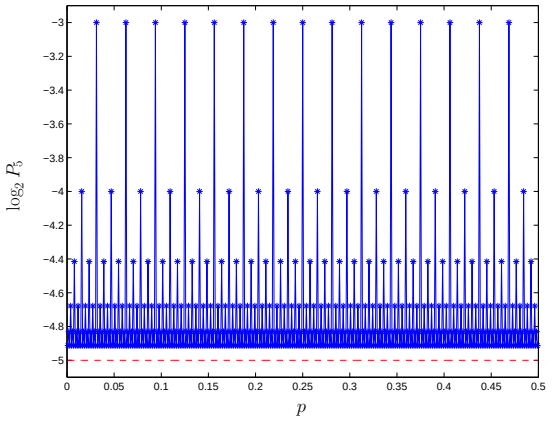


Figure 3.3: $\log_2 P_5$ versus p, where $n = 10, G_n(\cdot) = \text{floor}_n(\cdot)$
(The dashed line denotes $\log_2 \overline{\boldsymbol{P_5}} = -5$)

Remark 3.2: *Theorem 3.3 tells us: for the control parameters p with different resolutions (i.e., in different digital layers V_i), at least one values in $P_j(1 \le j \le n)$ will be different. In other words,* ***the resolution of p can be uniquely determined by the values of*** $\boldsymbol{P_1 \sim P_n}$.

In Figure 3.3, we give the experimental result of P_5 versus p when $n = 10$ and $G_n(\cdot) = \text{floor}_n(\cdot)$. To tell the truth, the strongly regular pattern shown in Figure 3.3 really astonished me when I plot them for the first time, at that time I have not proved the above theorems.

$P_j(1 \le j \le n)$ of the Digital Skew Tent Map (2.3)

For the digital skew tent map (2.3), we can easily get the following corresponding theorems similar to Theorem 3.2 and 3.3. Here, we omit the proofs.

Theorem 3.4: *Assume a discrete random variable x distributes uniformly in S_n. $\forall p \in V_i(1 \le i \le n)$, the following results are true for the digital skew tent map (2.3):*

1. *When* $G_n(\cdot) = \text{round}_n(\cdot)$, $P_j = \begin{cases} 2/2^j, & i \le j \le n \\ 1/2^{j-1}, & 1 \le j \le i-1 \end{cases}$;

   When $G_n(\cdot) = \text{floor}_n(\cdot)$ or $\text{ceil}_n(\cdot)$, $P_j = \begin{cases} 2/2^j, & i \le j \le n \\ 1/2^j + 1/2^i, & 1 \le j \le i-1 \end{cases}$;

2. $\forall k \in \{0, \cdots, 2^{n-i} - 1\}$, $P\{\text{floor}_{n-i}(F_n(x,p)) = k/2^{n-i}\} = 1/2^{n-i}$.

Corollary 3.4: *For the digital skew tent map (2.3), given two different control parameters $p \in V_i, p' \in V_{i'}$, where $i, i' = 1 \sim n$. We have: $i < i' \Leftrightarrow p \prec p'$.*

Theorem 3.5: *Assume a discrete random variable x distributes uniformly in S_n. $\forall p \in (0,1) \cap S_n$, the following results are true for the digital skew tent map (2.3):*

1. $\forall p \in D_i = S_i - \{0\} = \bigcup_{k=1}^{i} V_i$, $P_i = 2/2^i$;
2. $\forall p \in V_j (j \geq i+1)$, $P_i = \begin{cases} 1/2^i & , G_n(\cdot) = \text{round}_n(\cdot) \\ 1/2^i + 1/2^j & , G_n(\cdot) = \text{floor}_n(\cdot) \text{ or } \text{ceil}_n(\cdot) \end{cases}$.

§3.3.5 $P_j (1 \leq j \leq n)$ of $\mathcal{F}_n^k(x)$

From the discussion made in above subsections, we have known that a uniformly distributed digital signal will lead to non-uniform distribution after one chaotic iteration of a digital 1D PWLCM. Such non-uniformity will become more and more severe as the iterations go, i.e., the statistical properties of $\mathcal{F}_n^k(x)$ will become more and more non-uniform as k increases. Generally speaking, as k increases, $P_j (1 \leq j \leq n)$ will increase for most control parameters and sporadically decrease for some ones, and the regular pattern of P_j versus the control parameters and j will fade out slowly.

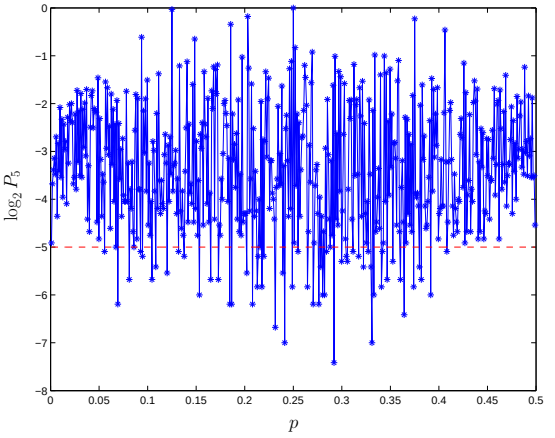


Figure 3.4: $\log_2 P_5$ of $\mathcal{F}_n^{32}(x)$ versus p
(The dashed line denotes $\log_2 \overline{P_5} = -5$)

In Figure 3.4, we give P_5 of $\mathcal{F}_n^{32}(x)$ versus p, where $\mathcal{F}_n(x)$ is the 1D PWLCM (2.1) and $n = 10, G_n(\cdot) = \text{floor}_n(\cdot)$. Comparing Figure 3.4 and Figure 3.3, we can see the value of P_5 increases at most control values and decreases at a small number of values, and at some values (for example, $p = 1/16$) it even reaches close to 1. The strong pattern in Figure 3.3 can never be discerned in Figure 3.4.

In my opinion, the reason of such an indistinct view should be attributed to the combination of the inherent complexity of continuous chaos and the dy-

namical degradation of digital chaos. Here, we will raise and try to answer the following question: Whether or not some rule exist in such an indistinct view dynamical indicators of $\mathcal{F}_n^k(x)$? Since the exact values of $P_j(1 \leq j \leq n)$ of $\mathcal{F}_n(x)$ can be strictly calculated, we think the results of $\mathcal{F}_n(x)$ may be extended to reflect the statistical properties of $\mathcal{F}_n^k(x)$ qualitatively.

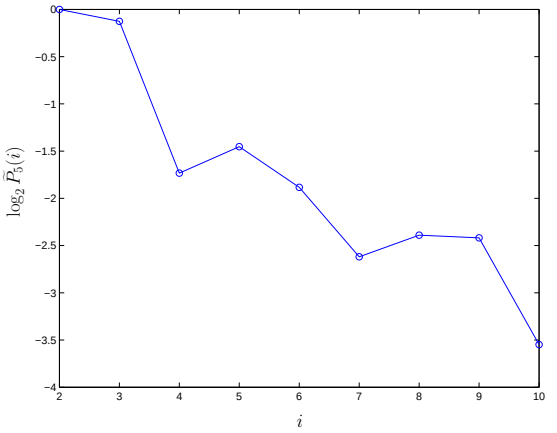


Figure 3.5: $\log_2 \widetilde{P}_5(i)$ of $\mathcal{F}_n^{32}(x)$

To simplify the discussion, let us consider the digital 1D PWLCM (2.1) as an example. From Corollary 3.3, we know that the weakest control parameter is $p = 1/4 \in V_2$, and the less weaker control parameters are ones in V_3, then those in V_4, V_5, $\cdots$, V_n. The above fact still **approximately** and **conceptually** holds for $P_j(1 \leq j \leq n)$ of $\mathcal{F}_n^k(x)$: use $\widetilde{P}_j(i)$ to denote the mean value of P_j of all control parameters with a same resolution i, we can see $\widetilde{P}_j(i)$ **roughly** decreases as i increases. For the data shown in Figure 3.4, $\widetilde{P}_5(i)$ is given in Figure 3.5. Yes, there really exists hidden order behind the chaotic sea.

§3.4 Applications of Dynamical Indicators

In this section, let us use see how can we use dynamical indicators in applications to find problems and enhance performances.

§3.4.1 A Performance Comparison of Different Remedies to Dynamical Degradation of Digital 1D PWLCM-s

In §2.5.2, three practical remedies to improve dynamical degradation of digital chaotic systems have been introduced: using higher finite precision[59, 64], cascading multiple chaotic systems[149], and (pseudo-)randomly perturbing the chaotic

systems[81, 82, 99, 170, 190, 195, 199, 203]. Our dynamical indicators can be used to qualitatively analyze performances of the three remedies in practice.

Using Higher Finite Precision

Considering the known scaling law of cycle length of pseudo orbits, in [59, 64] D. Wheeler suggested using higher precision to avoid the security problem about short cycle length of the chaotic key-stream in Matthews' chaotic stream cipher[58]. However, as we have mentioned in §2.5.1, there exist a large number of pseudo orbits whose lengths are much smaller than the mean length $O(2^{n/2})$ (recall the distribution of cycle periods). So using higher precision can only prolong the average cycle length of all chaotic orbits, not the cycle length of each chaotic orbit. That is to say, this remedy is not a good method to improve dynamical degradation of digital chaotic systems. Now let us use dynamical indicators of digital 1D PWLCM-s to re-discover this result.

From Eq. (3.19), we have known that $P_j = m \cdot \overline{\boldsymbol{P}}_j$ when $\max_{i=1}^{m}(r_i) \leq j \leq n$. We have mentioned that m can be used as an measurement of the dynamical degradation of a digital 1D PWLCM. In such a sense, higher precision cannot essentially improve the dynamical degradation at all if m is fixed. In addition, the following fact can also show the deficiency of using higher precision as a remedy to the dynamical degradation: higher precision cannot change the weakness of any control parameter in lower precision at all. For example, for the 1D PWLCM (2.1), $p = 1/4$ will always be absolutely weak for any precision, and $\forall p \in V_i$ will always be same weak for any precision $n \geq i$.

Consequently, assume the previous precision is n, using higher precision $n' > n$ can only improve the average performance of digital 1D PWLCM-s by introducing $n' - n$ new digital layers $V_{n+1} \sim V_{n'}$, but cannot improve the performance in S_n at all.

Cascading Multiple Chaotic Systems

In [149], the authors used two cascaded chaotic systems to increase the cycle length of generated chaotic orbits, where one chaotic system is used to initialize (control) another one every N iterations. Such a remedy can increase the length of the controlled pseudo orbit to $O(N)$ times. But it cannot enhance the non-uniformity of digital chaotic systems essentially, from our analyses on dynamical indicators in this chapter.

Consider k digital 1D PWLCM-s are cascaded, and the output of the i^{th} 1D PWLCM is used to initialize the $(i+1)^{th}$ 1D PWLCM every N_i iterations. Then the average cycle length of the whole system may be prolonged $O\left(\prod_{i=1}^{k-1} N_i\right)$ times.

Assume the input of the first 1D PWLCM distributes uniformly in S_n, we can know the output will not be uniformly distributed in S_n. Since the non-uniformly distributed output of the first 1D PWLCM is then used as the input of the second 1D PWLCM, the non-uniformity will become more serious. From such a viewpoint, k cascaded digital 1D PWLCM-s are composition of k same/different PWLCM-s, i.e., they will behave like $\mathcal{F}_n^k(x)$, which has been discussed in §3.3.5. As a summary, cascading multiple chaotic systems will make the dynamical properties of the final output more non-ideal, although it can effectively prolong the cycle length of the generated orbits.

The Perturbation-Based Algorithm

The perturbation-based algorithm is independently presented by J. Černák in [199] and Hong Zhou et al. in [170] as a practical tool to improve the dynamical degradation of digital chaotic systems. Tao Sang et al. generalized the algorithm to general cases in [81] and proposed an modified version in [82].

Here, for the sake of the reader's convenience, we briefly introduce the one proposed in [81] for discussion below. Given a simple PRNG with uniform distribution, run it to generate a small perturbing signal $\{S_p(i)\}$, which is then used to perturb the chaotic orbit $\{x(i)\}$ every Δ iterations, where Δ is a positive integer and the perturbing operation may be XOR or modular addition. There exist two available configurations, shown in Figure 3.6, we respectively call them Configuration A and B (A is suggested in [81, 82, 170] and B is suggested in [199]). Assume $\oplus$ denotes the perturbing operation, the two configurations can be expressed as follows:

$$\text{Configuration A} \quad : \quad x(i+1) = \mathcal{F}_n(x(i)) \oplus S(i), \tag{3.34}$$

$$\text{Configuration B} \quad : \quad x(i+1) = \mathcal{F}_n(x(i) \oplus S(i)), \tag{3.35}$$

where $S(i) = S_p(i/\Delta)$ if $i \bmod \Delta = 0$ and $S(i) = 0$ for any other i. The initial motivation of the proposal of perturbation is to prolong cycle lengths of pseudo orbits. It is obvious that the two configurations have similar performance on this point. But we will show Configuration A is better than B on another point.

Unlike the other two remedies, perturbation-based algorithm can also improve the non-uniformity of digital chaotic systems. In §3.3.5, we have pointed out the non-uniformity will become more and more serious as the chaotic system runs. Consider the perturbing signal exerted on chaotic orbits frequently smoothes the distribution of the orbits, such non-uniformity will be flatten every Δ iterations, which hints that the non-uniformity of perturbed chaotic systems will be approximate to the non-uniformity of $\mathcal{F}_n^\Delta(x)$. When $\Delta = 1$, the improvement will reach the best performance. Obviously, Configuration A has better

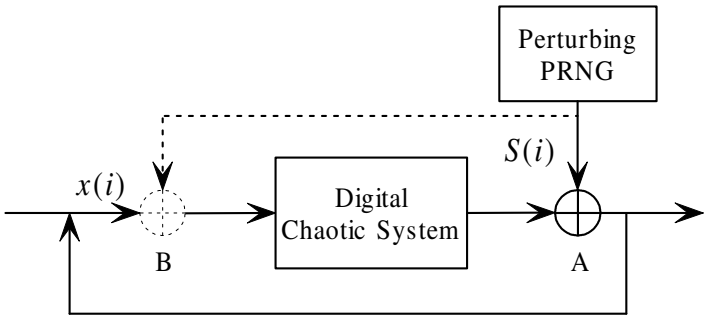


Figure 3.6: Two available configurations of the perturbation-based algorithm

performance on improving the non-uniformity than Configuration B, since the former smoothen both the input and output of the digital chaotic systems but the latter just smoothen the input. To sum up, the perturbation-based algorithm is a rather good method to practically improve the dynamical degradation of digital chaotic systems.

In [199], another different perturbing algorithm is also suggested, in which $S_p(i)$ is used to perturb the control parameter(s) of digital chaotic systems, not the pseudo orbits. Such an algorithm can also increase the cycle length, but cannot improve the non-uniform distribution efficiently enough. Consider the improvement on the non-uniformity is realized by mixing the non-uniformity of different control parameters, this algorithm has different performance for different initial control parameters: for the ones weaker than the mean level, such as $\forall p = 1/4 \in V_2$ of the digital 1D PWLCM (2.1), the non-uniformity may become better; for the ones stronger than the mean level, such as $\forall p \in V_n$ of the digital 1D PWLCM (2.1), the non-uniformity may become even worse. Based on such a fact, we can see the performance of this algorithm is even worse than Configuration B.

Although the perturbation-based algorithm can dramatically improve the dynamical properties of digital chaotic systems, there still exists dynamical degradation and some problems should be carefully considered to avoid potential defects in specific applications, especially in chaotic ciphers. Further discussion will be given in the following two subsections.

§3.4.2 Applications in Digital Chaotic Cryptography

As we know, digital 1D PWLCM-s have been widely used to construct digital chaotic ciphers[22, 24–26, 62, 63, 74, 77–79, 81, 82, 90, 96, 98, 106, 107, 110, 112, 116–118, 120]. The theoretical results about the proposed dynamical indicators $P_1 \sim P_n$ of digital 1D PWLCM-s are useful for the design and security analyses of such chaotic ciphers.

In §3.3, we have known that exact values of $P_j (1 \leq j \leq n)$ of a digital 1D PWLCM have tight relation with the resolutions of slopes of all linear segments. Also, it is possible to determine the resolutions of these slopes by observing values of the n dynamical indicators. Such a fact can be used to discern weak keys in some digital chaotic ciphers and develop some cryptanalytic methods.

In [24], Hong Zhou et al. presented a chaotic stream cipher based on the digital 1D PWLCM (2.1). The encryption procedure can be described as follows: use a maximal length LFSR to generate a pseudo-random signal $\{u_0(i) \in S_n\}$, which then is used to generate key-stream $k(i) = \mathcal{F}_n^k(u_0(i))$, where $\mathcal{F}_n(x)$ is realized in finite precision n and $k > n$. The perturbation-based algorithm proposed in [170] is used to enhance the dynamical degradation of $\mathcal{F}_n(x)$. The secret key is the control parameter p and the key space is $(0, 1/2) \cap S_n$.

From the results about $\mathcal{F}_n(x)$ we obtained in §3.3.4 and the practical performance of the perturbation-based algorithm, we can find there exist many weak keys that can be broken with less complexity than simple brute force attack. To facilitate the description here, let us assume the resolution of the secret key p is i. Then in known/chosen plaintext attacks, since the key-stream $k(t)$ is known, one can get i by observing n dynamical indicators $P_1 \sim P_n$. Of course, the perturbing signal in the last round should be removed to ensure the correctness of $P_1 \sim P_n$. Because the perturbation is public, such removal becomes natural and easy. Once i is known, one can search the secret key p in $(0, 1/2) \cap V_i$, whose size is smaller than the whole key space $(0, 1/2) \cap S_n$. From Theorem 3.3, it can be deduced that the expected number of known/chosen plaintexts is $O(2^i)$, since the difference between the largest $P_i = 4/2^i$ and the less largest $P_i = 2/2^i$ is large enough $(2/2^i)$ for distinguishing (see Fig 3.4). That is to say, the smaller i is, the faster p can be found and the weaker p will be. Extremely, several known/chosen plaintexts are enough to determine the weakest key $p = 1/4$. When the above idea is used to attack the chaotic cipher, we can calculate that the key entropy will decrease by 2 bits averagely. Experiments have been made to test the feasibility of this idea.

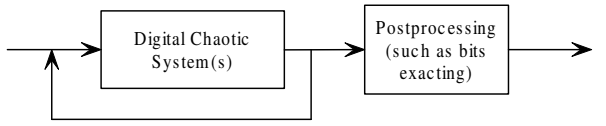
In fact, because of the similarity of another digital chaotic cipher proposed by Hong Zhou et al. in [25, 26], the above idea can also be available as a cryptanalytic tool. More details on such a cryptanalysis technique can be found in Chap. 4 and our paper [141]. Some possible remedies to enhance security of Hong Zhou et al.'s chaotic ciphers have been discussed in Chap. 4. Conceptually, all available remedies for Hong Zhou et al.'s ciphers can be extended to enhance security of other digital chaotic ciphers.

§3.4.3 Applications in Chaotic PRNG-s

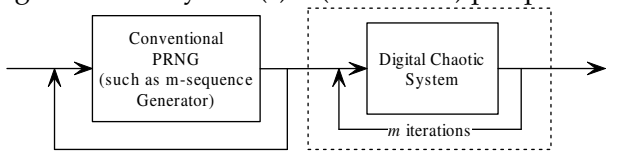
As we have mentioned in §2.2, many researchers have used digital 1D PWLCM-s to construct PRNG-s, and many ones are specially designed for the use in digital chaotic stream ciphers. Because of the non-uniformity of digital 1D PWLCM-s, pseudo-random numbers generated by digital 1D PWLCM-s will be not balanced. For example, if the digital 1D PWLCM (2.1) with $p = 1/4$ is selected and the lowest 2 bits of the chaotic orbits are used to generate pseudo-random bits, we can see they will be always zeros $000\cdots$ (recall Theorem 3.2 and Remark 3.1). In many chaotic PRNG-s, this problem is neglected.

To enhance the balance of the generated pseudo-random numbers, some remedy should be employed and the perturbation-based algorithm is still suggested since it can provide better performance than other two ones. Because there still exists non-uniformity even after perturbation, the stronger control parameters should play more positive role in chaotic PRNG-s than the weaker ones. If possible, we suggest only using the strongest control parameters, i.e., those in V_n.

In the following context, we will discuss two different structures of chaotic PRNG-s and explain the roles of digital 1D PWLCM-s in them. The two structures are shown in Figure 3.7a,b.



a) Digital chaotic system(s) + (Nonlinear) postprocessing



b) Conventional PRNG + Digital chaotic system

Figure 3.7: Two common structures of chaotic PRNG-s

The first structure (shown in Figure 3.7a) has been widely used in many chaotic stream ciphers and chaotic PRNG-s. In most cases, only a single digital chaotic system is used, but a couple are suggested by us in [22] to obtain pseudo-random numbers with higher security. The simplest version of this structure is the case when the unit linear transformation $f(x) = x$ is used for postprocessing, i.e., the chaotic orbit is directly output without any postprocessing. The most frequently used postprocessing method is bit-extracting algorithm: select limited (generally sequent) bits from the n-bit binary representation(s) of the chaotic or-

bit(s).

In secure applications of chaotic PRNG-s, if digital 1D PWLCM-s are used in the first structure with bit extracting post-process, we suggest extracting middle bits of the chaotic orbit(s) to generate pseudo-random numbers, which is because the following two facts: 1) the dependence of higher significant bits of the sequent chaotic states is somewhat larger than the one of lower bits*; 2) the dynamical degradation of digital 1D PWLCM-s chiefly exhibits on lower significant bits (recall Lemma 3.4) and the pseudo-random perturbation is mainly exerted on them. For example, if the 1D PWLCM (2.1) is used with the control parameter $p \in V_n$ and the chaotic orbit is represented as the format of $0.b_n b_{n-1} \cdots b_1, b_i \in \{0,1\}$, then $b_{\lfloor 2n/3 \rfloor} \cdots b_{\lceil n/3 \rceil}$ may be acceptable.

Another acceptable solution is to combine different bits of current state of the employed chaotic system. Generally, the combinations of different bits are strongly nonlinear operations, which can dramatically add the complexity of pseudo-random numbers without too much computation load. Also, accumulating multiple (and even all) previous states of the employed chaotic system to generate pseudo-random numbers can also provide much better performance. In [128], the above accumulating method is suggested by us to enhance security of Baptista's chaotic cipher.

The use of the second structure (shown in Figure 3.7b) can be found in [24]. In this structure, the digital chaotic system is used as a nonlinear postprocessing part of the conventional PRNG to enhance complexity of the pseudo-random numbers generated by the conventional PRNG (for example, enhance the linear complexity[145, 213] of m-sequence).

When digital 1D PWLCM-s are used in the second structure, the distribution of the pseudo-random numbers generated by the conventional PRNG will not be influenced much since digital 1D PWLCM-s have nearly uniform distribution. Thus, this structure can also be used in applications that require non-uniformly distributed pseudo-random numbers. Obviously, the digital chaotic system can also be considered as a smoothing filter with nonlinear transformation. In such a structure, if $m = 1$ or $\Delta = 1$, we can use $\text{floor}_{n-i}(\mathcal{F}_n(x))$ to generate nearly perfect pseudo-random output (recall Lemma 3.4 and the second result of Theorem 3.2). For example, assume the digital 1D PWLCM (2.1) are used here with $p \in V_{\lfloor n/2 \rfloor}$, the highest $n - \lfloor n/2 \rfloor$ bits of the final output of the chaotic PRNG will approximately preserve the original distribution of the pseudo-random numbers generated by the conventional PRNG. When stronger control parameters are used, some lower bits can also be output as a part of the generated pseudo-random

*Consider the following fact: If we know the highest $n/2$ bits of two sequent chaotic states $x(i+1) = F(x(i))$, it may be possible to **approximately** determine the control parameters of $F(\cdot)$; but we cannot find any useful information about the control parameter if we only know the lowest $n/2$ bits.

numbers. For example, $\forall p \in V_n$, the highest $\lceil 2n/3 \rceil$ may be OK. Experiments are required to practically determine the actual bit number.

§3.5 Conclusion

When chaotic systems are realized in a discrete space with finite states, the dynamical properties will far different from the ones described by the continuous chaos theory, and some degradation will arise. This problem plays important roles in the applications of chaotic systems in digital computers and circuits. In this chapter, we proposed a series of dynamical indicators of digital 1D PWLCM-s and detailedly investigate their calculations and applications for a class of digital 1D PWLCM-s with *onto* property (but the given analyses can be easily extended to general PWLCM-s). Theoretical results on the proposed dynamical indicators show that the digital chaotic output will not distribute uniformly when the input signal distributes uniformly in a discrete space S_n with finite precision n, and that the non-uniformity of the output signal can be quantitatively measured with n dynamical indicators: $1 \leq j \leq n$, $P_j = P\{\mathcal{F}_n(x) \in S_{n-j}\}$.

For other chaotic maps whose equations are not defined only by division, our results cannot straightforward be generalized. If some complicated mathematical functions with floating-point arithmetic are used in the equations, it will be much more difficult to find some measurable dynamical indicators and analyze their features for the studied digital chaotic systems, because floating-point decimals distribute in the discrete space with strongly non-uniform pattern*. If only chaotic iterations are made with floating-point arithmetic and all chaotic states are still stored as fixed-point numbers, the analysis may become easier.

In the future, it is a open topic to develop some available theoretical tools to analyze more digital chaotic systems. As the basis of possible solutions, some arithmetical models of different mathematical functions realized in finite precision (both with fixed-point and floating-point arithmetic) should be established firstly. For example, to analyze the piecewise nonlinear chaotic map proposed in [92], we should have a reasonable arithmetic theory of how $\sqrt{x}$ is calculated in digital computers.

*Such non-uniformity causes many well-known ill-conditioned problems in numerical algorithms, such as the entirely wrong solutions of some ill-equations numerically solved in finite precision.

Part II

Cryptanalyses of Some Recently-Proposed Digital Chaotic Ciphers

Chapter 4
Cryptanalysis of Hong Zhou et al.'s Chaotic Stream Ciphers

§4.1 Introduction

In 1996, U. Feldmann et al. proposed a general model for secure chaotic communications, which is called inverse system approach[76]. Soon Hong Zhou et al. pointed out some defects of inverse system approach, which make the encryption system not secure from the cryptographic point of view[25].

As a possible solution, Hong Zhou et al. suggested an enhanced chaotic encryption model of inverse system approach in [25, 26]. Different from the U. Feldmann et al.'s model, Hong Zhou et al.'s enhanced model is based on a kind of PWLCM realized in finite computing precision. Besides the above cryptosystem, Hong Zhou et al. also proposed some other chaotic stream ciphers based on PWLCM-s[24, 77–79].

Theoretically speaking, all Hong Zhou et al.'s chaotic ciphers are stream ciphers based on key stream generated from chaotic orbits of PWLCM-s. Hong Zhou et al.'s ciphers can be classified into two basic types: one type employs multiple chaotic iterations driven by a uniformly-distributed signal[24–26]; the other one is based on key stream generated from chaotic orbits filtered by a nonlinear map, which outputs different values when chaotic orbit goes into different subset of the whole phase space[77–79].

Till now, no any cryptanalytic work has been published on Hong Zhou et al.'s chaotic ciphers. The only related work was reported by Tao Sang et al. in 1999[92]. They pointed out that there may exist potential attacks to Hong Zhou et al.'s cipher in [78] since the employed chaotic map has piecewise linearity. Although they did't give any actual attacks, such a thought is not yet unreasonable since linear attacks can work in traditional cryptographical world[144, 145]. To avoid such a problem, they suggested using a class of piecewise nonlinear chaotic maps to replace PWLCM. It is obvious that Tao Sang et al.'s proposal is also suitable for other Hong Zhou et al.'s ciphers[77, 79].

In this chapter, we will try to give some cryptanalysis on the chaotic ciphers of Hong Zhou et al. proposed in [24–26].

Although Hong Zhou et al. have noticed dynamical degradation caused by digital chaotic maps realized in finite precision and proposed perturbation to solve this problem in practice[24, 77, 170], it is rather strange that they did not sug-

gest using perturbation in the chaotic ciphers proposed in [25, 26]. Apparently, for the above chaotic ciphers, dynamical degradation of digital chaotic systems cannot be neglected, since it destroys the uniform distribution of the key-stream and introduces many weak keys that cause large information leaking. In this chapter, aiming at Hong Zhou et al.'s chaotic ciphers in [25, 26], we will re-study security problems caused by digital chaos with a viewpoint more essential than Hong Zhou et al. gave in [170].

After discussing security problems caused by digital chaos, following the theoretical results on dynamical indicators in Chap. 3, we will point out that many weak keys exist in Hong Zhou et al.'s chaotic ciphers. Based on the weak-key analysis, we propose an enhanced brute force attack to lessen attack complexity of breaking related chaotic ciphers. In such an attack, the weaker the key is, the faster the attack will succeed. In a whole it can make the key entropy decrease by 2 bits. Although the key entropy decreases not much, the proposed attack is still useful since it reveals weak keys in the concerned ciphers (the weakest key is $p = 1/4$). To enhance Hong Zhou et al.'s chaotic ciphers, some possible solutions to weak keys are also discussed, and several ones seem helpful to enhance the security.

§4.2 Hong Zhou et al.'s Chaotic Ciphers

All concerned Hong Zhou et al.'s chaotic ciphers are based on the 1D PWLCM (2.1) extensively mentioned in previous chapters. See Figure 4.1 for its curve. In §3.2.1 we have known PWLCM-s with onto property has perfect dynamical

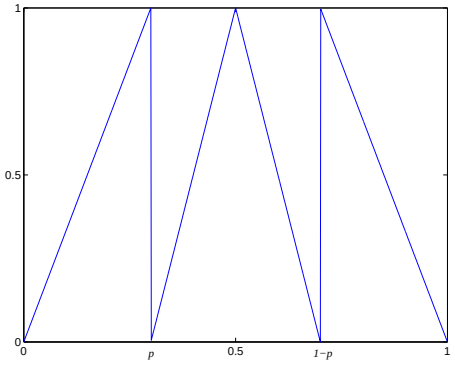


Figure 4.1: The PWLCM used in Hong Zhou et al.'s chaotic stream ciphers

properties. It is obvious that the PWLCM (2.1) is onto, so its dynamical properties in continuous space is desired to construct chaotic ciphers.

In [24], a n-order m-sequence $c(t)$ is used to generate the driven signal $u_0(t) = \sum_{i=1}^{n} 2^{-i} c(t+i-1)$, which is then taken as the initial condition of the above-mentioned PWLCM to generate the key stream $k(t) = u_k(t) = F^k(u_0(t), p)$. The ciphertext is obtained by XORing the plaintext with the key stream bit by bit like a normal stream cipher. To overcome potential security problems caused by dynamical degradation of digital PWLCM (2.1), perturbation algorithm[170] is suggested. Here, please note that m-sequence can be replaced by any other pseudo-random sequences with (pseudo-)uniform distribution.

The chaotic ciphers in [25, 26] are proposed to improve the security of previous inverse system chaotic encryption approaches. One typical cipher can be shown as follows:

$$\begin{aligned} &\text{Encryption: } y(t) = \Big[u(t) + F^k(y(t-1), p)\Big] \pmod 1, \\ &\text{Decryption: } u(t) = \Big[y(t) - F^k(y(t-1), p)\Big] \pmod 1, \end{aligned} \tag{4.1}$$

where $u(t)$ is the plaintext, $y(t)$ is the ciphertext and p is the secret key. As Hong Zhou et al. stated, the PWLCM (2.1) should be realized with n-bit finite precision. $n < k$ is needed to avoid recovery of the secret key p from some known/chosen plaintext/ciphertext pairs.

Actually, the ciphers in [25, 26] are stream ciphers with feedback of the ciphertext. Because the ciphertext $y(t)$ satisfies approximate uniform distribution in most cases, we can consider it as variants of the chaotic stream cipher in [24]: $y(t-1)$ in [25, 26] corresponds to $u_0(t)$ in [24].

In Fig. 4.2, we give a diagrammatic view of the three concerned chaotic ciphers, where DCS means "Digital Chaotic System". Method 1 denotes the chaotic cipher in [24] and Method 2 denotes the ones in [25, 26]. Here, please note that in [25, 26] Hong Zhou et al. did not suggest using perturbation algorithm, so perturbation only exists for Method 1.

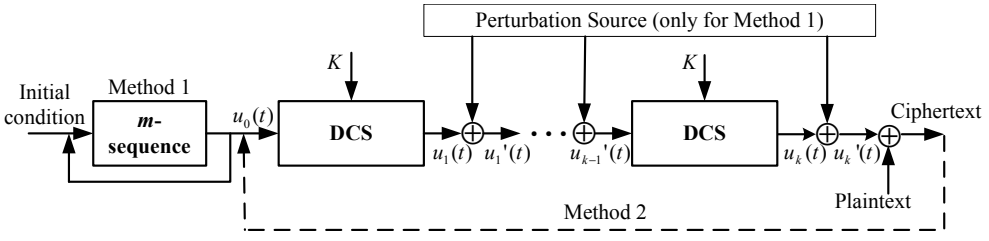


Figure 4.2: Chaotic stream ciphers proposed in [24–26]

In Chap. 3 we have shown that the dynamical indicators of the digital PWLCM (2.1) have measurable relation with resolution of the control parameter p. The use of perturbation cannot conceal such relation essentially, when the

perturbation algorithm is known by attackers. As a result, Hong Zhou et al.'s chaotic ciphers are not secure to known/chosen plaintext attacks because many weak keys exist. In the following sections, we will firstly make weak key analysis, and then introduce the enhanced brute force attack and show its performance. Finally, we will discuss some possible ways to mend the insecurity induced by weak keys.

§4.3 A Re-Study of Dynamical Degradation of the PWLCM (2.1) and its Negative Influence on Security of Hong Zhou et al.'s Ciphers

As we have suggested in §2.6.1, PWLCM-s are good candidates for digital chaotic ciphers because of their perfect dynamical properties and simple implementations in applications, if all known security problems about digital PWLCM-s are settled with proper remedies. Because Hong Zhou et al. did not suggest using any remedy to enhance the security of proposed chaotic ciphers in [25, 26], in this section we would like to re-study dynamical degradation of the digital 1D PWLCM (2.1) and investigate its negative influence on security of the proposed chaotic ciphers. This section can be considered as a concrete demonstration of my previous analyses.

Firstly, let us give a simple example to show how the dynamical degradation of the PWLCM (2.1) makes Hong Zhou et al.'s encryption scheme (4.1) insecure. Without loss of generality, following the definitions and notations used in §3.2.2, assume the finite precision $n = 8$ and $G_n(\cdot) = \text{floor}_n(\cdot)$. When $p = 3/8$, $y(t-1) = 1/16$, we can easily calculate that $F_n^9(y(t-1), p) = 0$. Since $k \geq n+1 = 9$, $F_n^k(y(t-1), p) = F_n^9(y(t-1), p) = 0$. Hence,

$$y(t) = \left\lfloor u(t) + F_n^k(y(t-1), p) \right\rfloor \bmod 1 = u(t). \tag{4.2}$$

That is to say, the plaintext $u(t)$ is directly output without encryption. Further experiments show that $y(t) = u(t)$ holds for 114 values in total 256 values of $y(t-1) \in S_n$, when $p = 3/8$. Such a great possibility of information leaking ($114/256 \approx 44.5\%$) will make the ciphertext-only attack and known-plaintext attack feasible. Thus, we can say $p = 3/8$ is a very weak key. Apparently, this serious problem is induced by the dynamical degradation of the digital PWLCM (4.1), since $P\{F^k(x, p) = 0\} = 0$ for the real-valued version of this map (it is ergodic[210]).

Then let us investigate how many weak keys there are in Hong Zhou et al.'s encryption scheme (4.1) under n-bit finite precision. Rigorously, for a secret key

p, if the probability of $y(t) = u(t)$ is larger than 2^{-n}, it can be regarded as a weak key. The larger the probability is, the weaker the key will be. To measure the weakness level of a given key p, define the *weak factor* $\alpha(n, k, p)$ as follows:

$$\alpha(n, k, p) = P\left\{F_n^k(y(t-1), p) = 0\right\} \Big/ 2^{-n}. \tag{4.3}$$

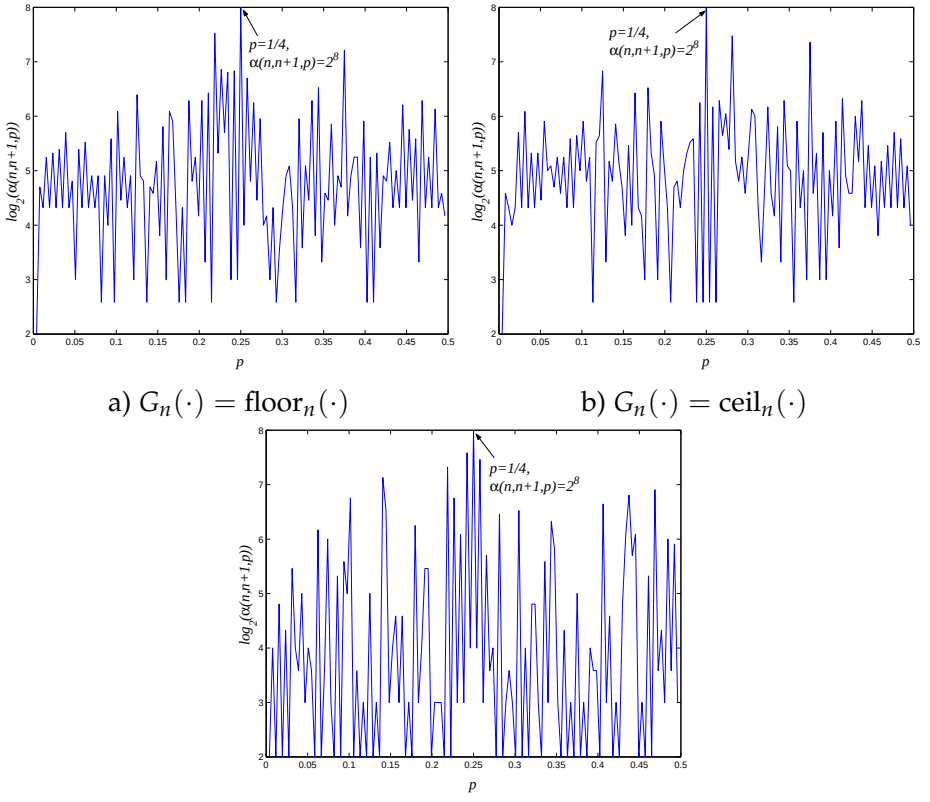
Here, $\alpha(n, k, p) > 1$ indicates p is a weak key; and the larger $\alpha(n, k, p)$ is, the weaker the key will be. In addition, when $F_n^k(y(t-1), p)$ distributes uniformly in S_n, $\alpha(n, k, p) = 1$ (i.e., $P\left\{F_n^k(y(t-1), p) = 0\right\} = 2^{-n}$), so $\alpha(n, k, p)$ also can partially reflect the non-uniformity of the distribution of $F_n^k(y(t-1), p)$ in S_n. From (4.3), we can easily get $k_1 > k_2 \Rightarrow \alpha(n, k_1, p) \geq \alpha(n, k_2, p)$, which means that $\alpha(n, n+1, p)$ is the lower bound of $\alpha(n, k, p)$ for all values of k. Thus, in the following context, we assume $k = n + 1$ to make the experiments (in fact, $k = n + 1$ is also the optimal value of Hong Zhou et al.'s cipher since larger k implies heavier computation load).

When $n = 8$ and $G_n(\cdot)$ is respectively $\text{floor}_n(\cdot)$, $\text{ceil}_n(\cdot)$ and $\text{round}_n(\cdot)$, Figure 4.3 gives the values of $\log_2(\alpha(n, n+1, p))$ for different keys. From the experimental data, we can find the following facts:

- **Fact 1**: $\alpha(n, n+1, p) > 1$ almost everywhere, and many keys are rather weak since $\alpha(n, n+1, p) \gg 1$.
- **Fact 2**: The weakest key is $p = 1/4$, which makes $\alpha(n, n+1, p) = 2^8$ so that $P\{y(t) = u(t)\} = 1$ (the cipher disappears!).
- **Fact 3**: The number of weak keys when $G_n(\cdot) = \text{round}_n(\cdot)$ is less than the numbers when $G_n(\cdot) = \text{floor}_n(\cdot)$ and $G_n(\cdot) = \text{ceil}_n(\cdot)$, so $\text{round}_n(\cdot)$ can provide better security than $\text{floor}_n(\cdot)$ and $\text{ceil}_n(\cdot)$. This fact is natural since $\text{round}_n(\cdot)$ can introduce smaller quantization errors than $\text{floor}_n(\cdot)$ and $\text{ceil}_n(\cdot)$ in general.

Apparently, the above facts show that Hong Zhou et al.'s chaotic cipher (4.1) is not secure enough from strict cryptographic viewpoint.

Recall the dynamical indicators we introduced in Chap. 3, it is obvious that $\alpha(n, 1, p) = P_n/2^{-n}$. Actually, with the theoretical results proved in §3.3.4, the above experimental results about $\alpha(n, n+1, p)$ can be qualitatively explained. As a typical PWLCM showing dynamical degradation, we have proved their dynamical indicators $P_j = P\{F_n(x, p) \in S_{n-j}\} (j = 1 \sim n)$ satisfy Theorem 3.2 and 3.3. The two theorems show that the dynamical degradation of the digital PWLCM (4.1) can be measured by the *resolution* of the control parameter p. Generally speaking, the smaller the resolution of p is, the more serious the dynamical degradation will be. Following such a statement, the weakest key will be $p = 1/4$



a) $G_n(\cdot) = \text{floor}_n(\cdot)$ b) $G_n(\cdot) = \text{ceil}_n(\cdot)$

c) $G_n(\cdot) = \text{round}_n(\cdot)$

Figure 4.3: The weak factor $\log_2(\alpha(n, n+1, p))$ versus p when $n = 8$

since it has the smallest resolution 2, which agrees with the above Fact 2 obtained from the experimental data in Fig. 4.3.

Of course, because $k > n > 1$, the rigorous result about P_j and the resolution of p cannot simply extend to explain the relation between $\alpha(n, k, p) = P\{F_n^k(x, p) = 0\}/2^{-n}$ and p. Observe Fig. 4.3, besides the control parameters with small resolutions, we can see some ones with large resolutions also become very weak, such as $p = 29/128$ and $p = 31/128$ (both with the resolution of $n - 1 = 7$). It means the dynamical degradation of digital chaotic systems will become more and more serious and complicated as k increases, which leads to the qualitative results given in §3.3.5.

Now we reach the conclusion that the digital chaotic ciphers proposed in [25, 26] are not secure without remedies to improve the dynamical degradation of the digital PWLCM (2.1). Naturally, we can use the perturbation algorithm suggested in [170] to avoid this problem. However, in the next section, we will

show that there are still some security problem in such enhanced chaotic ciphers with perturbation algorithm.

§4.4 Weak-Key Analysis and an Enhanced Brute Force Attack

From §3.2.1, the PWLCM (2.1) has uniform invariant density function $f^*(x) = 1$, which means the following fact: if the input signal $u_0(t)$ distributes uniformly in the definition domain $I = [0,1]$, then the output signal of the PWLCM $u_1(t) = F(u_0(t), p)$ will also distribute uniformly in I. This feature is the security basis of Hong Zhou et al.'s chaotic ciphers in [24–26]. However, as we have proved in §3.3.4, when such a PWLCM is realized with finite precision in digital circuits or computers, dynamical degradation will occur and such degradation is measurable with n dynamical indicators.

Now let us recall the results given in Chap. 3. When the PWLCM (2.1) is realized in n-bit finite precision, if the input signal $u_0(t)$ yields to discrete uniform distribution, the output signal $u_1(t) = F(u_0(t), p)$ does not hold discrete uniform distribution. The departure of $u_1(t)$ from discrete uniform distribution can be described as follows with the n dynamical indicators $P_1 \sim P_n$: 1) $\forall p \in S_n$, $P_j > 1/2^j$; 2) the value of P_j is uniquely determined by the resolution of p (Theorem 3.2 and 3.3); 3) the maximal value of P_j is $4/2^j$, and the second maximal value is $2/2^j$. The third item shows that it is possible for us to get the resolution of p (i.e., to get which digital layer the key p is in) by observing n probabilities $P_1 \sim P_n$. This fact opens a back door to find weak keys and lessen complexity of brute force attack.

As we described in §4.2, the three analyzed chaotic ciphers have similar structures, so in this section we will only focus our cryptanalysis on the typical chaotic cipher proposed in [24]. In the following contexts, we will use the notations described in §3.2.2 to make the description more clearer.

For the PWLCM (2.1), considering $0 < p < \frac{1}{2}$, when it is realized in n-bit finite precision, the key space is $S_n \cap (0, \frac{1}{2})$ and the key entropy under simple brute force attack is $K = \log_2(2^{n-1} - 1)K \approx n - 1$. To facilitate the following description, define $S'_i = S_i \cap (0, \frac{1}{2}), V'_i = V_i \cap (0, \frac{1}{2})$, and define the complete multi-resolution decompression of S'_n as $\{V'_i\}_{i=2}^n$, where $S'_n = \bigcup_{i=2}^n V'_i$, $\forall i \neq j, V'_i \cup V'_j = \varnothing$ (the decompression level is $n - 1$).

§4.4.1 Weak-Key Analysis

Making complete multi-resolution decompression on S'_n, we can get $S'_n = \bigcup_{i=2}^n V'_i$. From Theorem 3.3 (considering $V'_i \subset V_i$), for two secret keys $p \neq q$

in different digital layer sets $V'_i, V'_j (i \neq j)$, there exist (distinguishable) difference between their dynamical indicators $P_{i_1} \sim P_{i_n}$ and $P_{j_1} \sim P_{j_n}$. Therefore, it is possible to get the resolution of the secret key when its n dynamical indicators $P_1 \sim P_n$ are all known. Once we get the resolution of p, we can exhaustively search the key only in a subset of the whole key space, which will lessen attack complexity. The smaller the resolution i is, the smaller the subset V'_i will be (2^{i-2} possible keys) and the faster the attack will succeed. In other words, the smaller the resolution is, the weaker the key will be.

Then the problem becomes: how can we get the values of $P_1 \sim P_n$? Because of the special structure of Hong Zhou et al.'s ciphers, it is possible to observe $P_1 \sim P_n$ under known/chosen-plaintext attack scenario. To facilitate the following explanation, assume the i^{th} output of the PWLCM is $u_i(t)$, and the i^{th} perturbed output is $u'_i(t) = u_i(t) \oplus pt_i(t)$ (where $pt_k(t)$ is the i^{th} perturbing signal), then the key stream $k(t) = u'_k(t)$ (see also Figure 4.2). When the plaintext $P(t)$ and ciphertext $C(t)$ are both known, $k(t)$ will be also known to be $P(t) \oplus C(t)$. Since the details of perturbation algorithm is also public, we can remove the last round perturbing signal $pt_k(t)$ to get the last chaotic output $u_k(t) = k(t) \oplus pt_k(t)$. Consider $u_k(t) = F(u'_{k-1}(t), p)$ and the $k-1^{th}$ perturbed chaotic output $u'_{k-1}(t)$ satisfies approximate discrete uniform distribution in S'_n[170], $u_k(t)$ yields to Theorem 3.2 and 3.3, that is to say, the values of $P_1 \sim P_n$ can be estimated by observing $u_k(t)$. As the number of known/chosen plaintexts increases, the estimated values of $P_1 \sim P_n$ will probabilistically converge at the theoretical values given in the above two theorems. Since there exists enough difference between the maximal value and the second maximal value of each P_j ($4/2^j - 2/2^j = 2/2^j$), p's resolution will be distinguished with a number of plaintext/ciphertext pairs.

To show the existence of weak keys in the Hong Zhou et al.'s chaotic ciphers and how to find their resolutions, now let us consider the weakest key in the whole key space: $p = 1/4 \in V'_2$. From Theorem 3.2, $P_2 = 1$, which means the least two bits of $u_k(t)$ are always zeros. When $p = 1/4$ is selected as the secret key, we can quickly find the fact that $P_2 = 1$ (the second maximal value of P_2 is $1/2$) by observing $u_k(t)$ and then directly get $p = 1/4$. When $p \in V'_3$, $P_2 = P_3 = 1/2$, the deduction procedure is a little more complex: calculate the estimated values of P_2 and P_3 to determine $p \in V'_3$ - use $P_3 = 1/2 = 4/2^3$ to determine $p \in S'_3$ and use $P_2 = 1/2 < 1 = 4/2^2$ to determine $p \notin S'_2$, and then exhaustively search the right key in V'_3 (total $2^{3-1} - 2^{2-1} = 2$ possible keys: $1/8, 3/8$). Similarly, when $p \in V'_i$, values of P_{i-1} and P_i are required to distinguish the resolution i, and the number of total keys in V'_i is $2^{i-1} - 2^{i-2} = 2^{i-2}$. The above description on weak keys can be summarized in Table 4.1.

Apparently, when Hong Zhou et al.'s chaotic ciphers are realized in n-bit

Table 4.1: A comparison of secret keys with different resolutions

resolution of p	2	3	$\cdots$	i	$\cdots$
observed probabilities	P_2	P_2, P_3	$\cdots$	P_{i-1}, P_i	$\cdots$
required plaintexts	$O(2^2)$	$O(2^3)$	$\cdots$	$O(2^i)$	$\cdots$
sub-key-sapce	V_2'	V_3'	$\cdots$	V_i'	$\cdots$
size of sub-key-sapce	1	2	$\cdots$	2^{i-2}	$\cdots$

finite precision, the whole key space can be divided into $n-1$ sub-spaces $V_i'(2 \leq i \leq n)$, whose cryptographical strengths exponentially decreases as i decreases from n to 2. As a natural result, if we increase the current computing precision from n to n', $n'-n$ new sub-space(s) $V_{n+1}' \sim V_{n'}'$ will be introduced, but all weak keys in previous precision n will not become stronger at all*.

From the above discussion, the basic procedure to determine the resolution of p can be described as follows. For each known/chosen plaintext, calculate $u_k(t)$ and then obtain the current estimated probabilities $P_2 \sim P_n$ (P_1 is neglected since V_1 is not contained in S_n'). When each P_i becomes stable within a small range, the resolution i can be determined by the following rule: if P_i converges at the maximal value $4/2^i$ and P_{i-1} converges at the second maximal value $2/2^{i-1}$, then the resolution is i. Probabilistically, the number of known/chosen plaintexts is $O(2^i)$. From the number of required plaintexts, we can see the smaller p's resolution i is, the faster it will be to obtain i through observing $P_2 \sim P_n$ and the weaker p will be. Later in §4.5, we will give some actual experiments to show correctness and feasibility of such a weak-key analysis.

§4.4.2 An Enhanced Brute Force Attack

From the discussion in the above subsection, we can easily propose an enhanced brute force attack, which is a known/chosen plaintext attack and has less attack complexity than simple brute force attack. In the proposed attack, firstly the secret key's resolution i is estimated from a number of known/chosen plaintext/ciphertext pairs, and then the exhaustive searched is made in the sub-space V_i' to find the right key. A C-language description of the proposed attack is as follows:

- **Requirements**: known/chosen plaintexts and corresponding ciphertexts, and public details of perturbation algorithm, i.e., known $u_k(t)$;
- **Variables**: `u_k[t]` – $u_k(t)$, `P[i]` – the estimated value of P_i, `Pn[i]` – the occurrence number of $u_k(t) \in S_{n-i}$, `e1[i]` – a threshold to determine `P[i]`

*This fact makes the remedy using higher precision a passive countermeasure, as we discussed later.

is approximately the maximal value of P_i, `e2[i-1]` – another threshold to determine `P[i-1]` is approximately the second maximal value of P_{i-1};

- **Initialization**: `for(i=2;i<=n;i++) {Pn[i]=0; P[i]=0;}`
- **Determining the secret key's resolution *i***:

```
for(t=0;;t++)
{   for(i=2;i<=n;i++)
    {   temp=floor(u_k[t]*pow(2,n));
        if((temp<<(n-i))==0)
        { Pn[i]++; P[i]=Pn[i]/t; }
    }
    for(i=2;i<=n;i++)
    {   if(fabs(P[i]-4/pow(2,i))<=e1[i]
        && fabs(P[i-1]-2/pow(2,i-1))<=e2[i-1])
            goto end;
    }
}
end: printf("The resolution is %d.",i);
```

There are some remarks on the above procedure:

- **Remark 1**: Please note that translating `u_k[i]` into an integer needs the use of floating-point multiplications, which are time-consuming operations. In fact, since the decimals in [24–26] are all represented with fixed-point arithmetic, we can consider them as integers and then the floating-point multiplications become fast left-shift operations.
- **Remark 2**: Because $u_k(t)$ is approximately (not strictly) yield to the requirements in Theorem 3.2 and 3.3, there exist a small different between the estimated values and the theoretical values. Therefore, the threshold values `e1[i]` and `e2[i-1]` are required to make the attack robust. The selection principles of the two thresholds are $0 <$`e1[i]`$< 1/2^i$ and $0 <$`e2[i-1]`$< 1/2^{i+1}$, and their actual values can be determined with experiments.
- **Remark 3**: In [24], m-sequence is used to generate $u_0(t)$, so $u_0(t)$ is always positive (not be zero forever), which will influence the correspondence between the actual probabilities and the theoretical ones: when $u_0(t) = 0, \forall i \in [1, n]$, $F_n(u_0(t), p) = 0 \in S_{n-i}$, so $u_0(t) \neq 0$ will make the actual probabilities less than the theoretical ones. Fortunately, because of the use of perturbation, $u'_{k-1}(t)$ has approximate uniform distribution in S_n, the problem of $u_0(t) \neq 0$ and the above influence are

avoided. For ciphers in [25, 26], the ciphertext $y(t-1)$ approximately yields to uniform distribution and there does not exist such a problem.

- **Remark 4**: There is an equivalent method to judge whether or not `P[i-1]` approximately goes to the second maximal value of P_{i-1}: just to judge whether or not `P[i-1]` is obviously less than the maximal value of P_{i-1}. With such a method, the threshold `e2[i-1]` can be omitted or its value can be relaxed.
- **Remark 5**: To avoid getting wrong resolution, we can introduce a stable indicator η. Only when `P[i]` and `P[i-1]` continuously yield to desired requirements for at least η times, the found resolution i is returned. Although η will make the number of required known/chosen plaintexts increase (generally not much), it is helpful to promote the robustness of the proposed attack.

- **Searching the sub-space V_i'**: Search the right key in V_i' with a known plaintext/ciphertext pairs. The average number of searched keys is $2^{i-2}/2 = 2^{i-3}$.

§4.4.3 Performance of the Enhanced Brute Force Attack

In this subsection, let us make the performance analysis of the proposed enhanced brute force attack, where we assume that the key is uniformly selected at random from the whole key space S_n' (a reasonable assumption in practice).

1. **The average number of known/chosen plaintexts**:
   $N_p = \sum_{i=2}^{n} \frac{2^{i-2}}{2^{n-1}-1} \cdot 2^{i-2} = \frac{2^{2n-1}-1}{3 \cdot (2^{n-1}-1)} \approx \frac{2^{n-1}}{3}$.
2. **The average number of searched keys (search complexity)**:
   $N_k = \sum_{i=2}^{n} \frac{2^{i-2}}{2^{n-1}-1} \cdot 2^{i-3} = \frac{2^{2n-2}-1}{6 \cdot (2^{n-1}-1)} \approx \frac{2^{n-2}}{3}$.
3. **Key entropy under the proposed attack**:
   $H(K) = \sum_{i=2}^{n} \frac{2^{i-2}}{2^{n-1}-1} \cdot (i-2) = \frac{(n-3) \cdot 2^{n-1}+2}{2^{n-1}-1} \approx n-3$, which is less than 2 bits than $n-1$ (the key entropy under simple brute force attack).

From the above results, we can see that the overall performance of the proposed attack is better than simple brute force attack, but the average number of known/chosen plaintexts is a little large and the improvement of the key entropy is not so satisfactory as expected. The significance of the proposed attack lies in the following two facts: 1) the proposed attack can **effectively** break the chaotic

ciphers when the selected keys are VERY weak*; 2) the security problem caused by dynamical degradation of digital chaotic systems shows the difficulty to make a digital chaotic cipher really secure.

§4.5 Experiments and Simulations

To confirm the theoretical analysis made in this chapter, we make some actual experiments to test the results obtained in the last section. Experimental data agree with our theoretical analysis and show the feasibility of the weak-key based attack. The parameters of the chaotic cipher are selected as follows:

- The finite computing precision $n = 10$ (as Hong Zhou et al. did in [24], low precision is used to simplify the experiments and to make statistics of corresponding ciphers practically possible);
- The DATF is $\text{floor}_n(\cdot)$;
- The number of chaotic iterations $k = n + 1 = 11$;
- The primitive polynomial of the driven m-sequence[214] is $1 + x^3 + x^{10}$ and the initial state is 1;
- A m-sequence generator is used to generate the perturbing signal, its primitive polynomial is $1 + x^2 + x^3 + x^8 + x^{10}$ and its initial state is 1;
- The number of perturbed bits $n_m = 5$.

§4.5.1 Performance of Perturbation

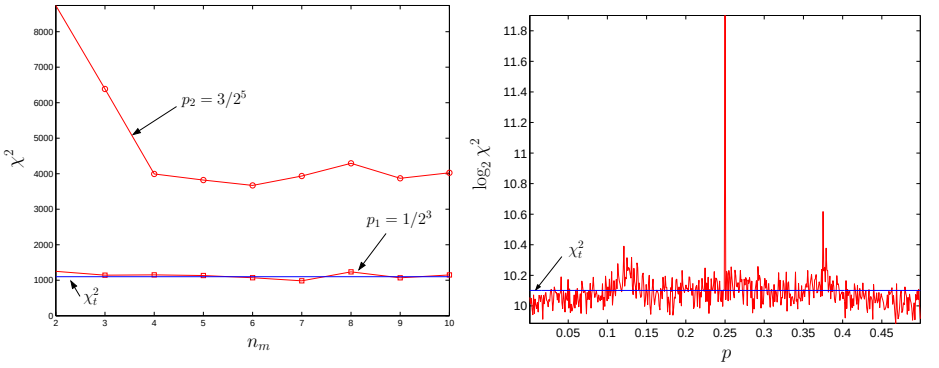
A basic requirement of the existence of weak keys is that $u'_{k-1}(t)$ approximately satisfies uniform distribution. At first, some experiments are made to confirm the results given in [24, 170] (i.e., to see whether or not $u'_{k-1}(t)$ yields to such a requirement).

Hong Zhou et al. gave a conceptual rule $n_m = \lceil \log_2(k_{max} + 1) \rceil$ to determine the number of perturbed bits in [170], but the estimated value is always relatively larger and is different for different key (which make the implementation of perturbation algorithm more complex). So we re-study this issue via our experiments. We found the following facts: 1) perturbing a small number of bits is enough to effectively improve the uniformity of the distribution of the chaotic output; 2) when n_m is beyond a threshold value, the efficiency of perturbation

*Generally speaking, the secret keys with resolution not greater than $n/2$ are rather weak. From the strictest viewpoint, only the key with the resolution n are not weak, but the number of such strong keys is only half of the total number.

become trivial; 3) the weaker the key is, the improvement of uniform distribution with the same n_m will be less better. When $n = 10$, $n_m = 5$ can reach satisfactory performance.

For $n = 10, k = 11$, we make Pearson-χ^2 test on $u'_k(t)$[212]. The test parameters are: the significance factor $\alpha = 5\%$, the number of intervals $m = 1024$ and the refuse threshold $\chi^2_\alpha(m-1) = 1098.5$. When the key $p = 1/2^3, 3/2^5$, the performances of different n_m are shown in Figure 4.4a; when $n_m = 5$, the performances of different keys are shown in Figure 4.4b.



a) Perturbation performance of different values of n_m

b) Perturbation performance of different keys when $n_m = 5$

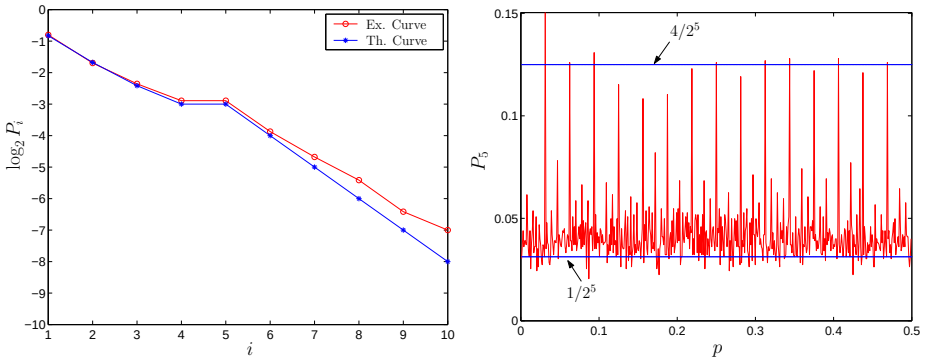
Figure 4.4: Pearson-χ^2 test of $u'_k(t)$

§4.5.2 The Estimated Values of $P_2 \sim P_n$

Since $u'_{k-1}(t)$ approximately satisfies uniform distribution in S'_n, from Theorem 3.2 and 3.3, the estimated values of $P_2 \sim P_n$ calculated from $u_k(t)$ should converge at the theoretical values. Of course, because the actual distribution of $u'_{k-1}(t)$ is not strict uniform distribution, there still exists small differences between `P[i]` ($i = 2 \sim n$) and the theoretical values. Our experiments show that the differences are small enough to ensure distinguishing of the secret key's resolution i.

In Figure 4.5, we give the comparison of the estimated values from $u_k(t)$ and the theoretical values. Comparing the two sub-figures 4.5a and b with Figure 3.1a and Figure 3.3, we can see the correspondence between the estimated values and the theoretical ones are satisfactory. Figure 4.5a shows that when $p = 3/2^5$ the experimental curve corresponds to the theoretical one with acceptable bounds (please note that here we use logarithmic coordinate to enlarge the visual difference). When i is close to n, compared with smaller i, the deviation becomes larger,

which is because P_i with larger i is more sensitive to the nonuniformity of $u'_{k-1}(t)$ than P_i with smaller i (consider $i < j \rightarrow P_i > P_j$). Figure 4.5b shows that there are enough difference between the maximal value and the second maximal value of P_5, which is needed to distinguish the resolution of the secret key. Although there are fluctuations in the values of P_5 of all keys in V'_5, the fluctuations are not so serious to influence the distinguishability of the keys' resolutions. The existence of fluctuations is the direct reason why we must use two thresholds `e1[i]` and `e2[i-1]` to judge whether or not `P[i]` approximately equals to the maximal value or the second maximal value.



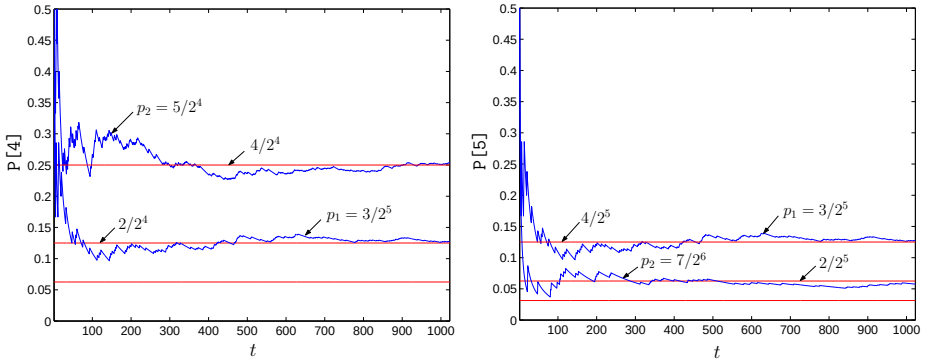
a) the estimated values of $P_1 \sim P_n$ when $p = 3/2^5$

b) the estimated values of P_5 for different keys

Figure 4.5: The comparison of the estimated values of $P_1 \sim P_n$ from $u_k(t)$ and the theoretical values

§4.5.3 An Actual Attack

Finally, assume the secret key $p = 3/2^5 \in V'_5$, we give an actual attack on Hong Zhou et al.'s cipher in [24]. Figure 4.6a and 4.6b respectively give the changing curves of `P[4]` and `P[5]` as the number of known/chosen plaintexts increases. We can see after a short period `P[4]` and `P[5]` gradually converges at the theoretical values $2/2^5$ and $4/2^5$ (with a small fluctuations). Once `P[5]` and `P[4]` goes into the acceptance range defined by the theoretical values and the two thresholds `e1[5]`,`e2[4]` for η times, the resolution $i = 5$ can be determined and the search of the key in the sub-space can start. The number of observed plaintexts is about $O(2^5)$. As a comparison, the changing curve of `P[4]` when $p = 5/2^4 \in V'_4$ is given in Figure 4.6a, and the curve of `P[5]` when $p = 7/2^6 \in V'_6$ is given in Figure 4.6b. It can be observed that curves of keys with different resolutions gradually depart as the number of known plaintexts increases.



a) `P[4]` versus t (the number of observed plaintexts): $p = 3/2^5, 5/2^4$

a) `P[5]` versus t (the number of observed plaintexts): $p = 3/2^5, 7/2^6$

Figure 4.6: An actual attack when the secret key $p = 3/2^5$

§4.6 Discussion on Possible Remedies

To avoid the security problem caused by weak keys and the proposed attack, in this section we will discuss some possible remedies and their performances. Many remedies can also be used as useful principles to enhance security of other digital chaotic ciphers. For each remedy, I use a word "Yes", "No" or "Uncertain" in the title to show our opinion on its use in practice.

§4.6.1 Using Higher Finite Precision: No

It seems that increasing the computing finite precision is the simplest and the most convenient method to enhance the security of a digital chaotic cipher. From the analysis shown in the above subsection, we can see higher precision can make the key entropy become larger. However, as we have known in §4.4.1, when we increase the computing precision from n to n', the improvement of the overall security is realized by newly-introduced $n' - n$ sub-spaces and all weak keys in previous n-bit precision cannot be improved at all.

What's more, there are some clues to show higher precision even makes things worse. When the perturbation algorithm is not used, experiments show that the weak keys analyzed in §4.3 cannot be improved rapidly as the finite precision n increases. In Figure 4.7, some results are given for $p = 3/8$, $1/16$ and $13/64$ when $n = 6 \sim 19$. We can see that $\alpha(n, n+1, p)$ becomes larger and larger in general as n increases.

So we think using higher finite precision is not an essential remedy to related chaotic ciphers.

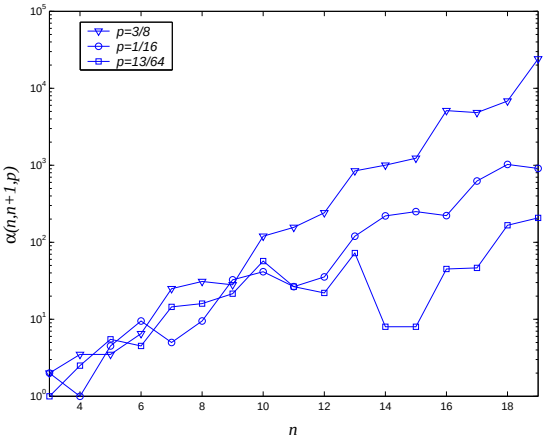


Figure 4.7: $\alpha(n, n+1, p)$ versus $n = 6 \sim 19$ (logarithmic Y-axis is used)

§4.6.2 Employing More Complex Chaotic Systems: Uncertain

As we know, the existence of weak keys in Hong Zhou et al.'s cipher depends on the dynamical degradation of the employed PWLCM in finite computing precision. Then how about using other chaotic systems to realize the chaotic ciphers to enhance the security? Maybe it is an essential solution. Of course, the employed chaotic systems should satisfy the good dynamical properties of PWLCM (2.1). One candidate chaotic system is the piecewise nonlinear chaotic map proposed and analyzed in [92]:

$$F(x) = \begin{cases} \frac{1}{a_i}\left(\sqrt{4a_i\left(\frac{x-c_i}{c_{i+1}-c_i}\right) + (1-a_i)^2} - 1\right), & x \in [c_i, c_{i+1}) \\ 1, & x = 1 \\ F(-x), & x \in [-1, 0) \end{cases}, \tag{4.4}$$

where $0 = c_0 < c_1 < \cdots < c_N = 1$, $a_i \in (-1, 0) \cup (0, 1)$ and $\sum_{i=0}^{N-1}(c_{i+1} - c_i) \cdot a_i = 0$. It has been proved that the above maps also have uniform invariant density functions $f(x) = 0.5$[92], which is a significant property of the PWLCM (4.1) used in Hong Zhou et al.'s encryption scheme. Is such a map OK? It is rather hard to give the right answer. It has been widely known that dynamical degradation of the digital PWLCM (2.1) also exists in other digital chaotic maps (recall §2.5). Thus, essentially speaking, the dynamical degradation cannot be avoided for any digital chaotic system. Because there is not yet an established theory to measure the exact dynamical properties of digital chaotic systems, it is rather difficult to select a "really" better chaotic map than the PWLCM (2.1). In addition, the use of more complex chaotic systems will increase the computation

load, influence encryption speed badly and increase the implementation costs. Before more convincing theoretical results are given, we must be very careful to make decisions.

§4.6.3 Keeping Perturbation Parameters Secure: Yes?

As we have shown before, only when details about the perturbation algorithm is known by attackers, it is possible for them to get $u_k(t)$ from the key stream (i.e., the last chaotic output) $k(t) = u'_k(t)$ and then calculate $P_2 \sim P_n$ to get the secret key's resolution i.

Thus, if we conceal details on the perturbed algorithm, the security of the chaotic cipher may be improved. From Kerckhoffs' principle[144, 145], the perturbation algorithm itself should not be secure, and we have to take the seed, the parameters of the perturbing PRNG and the number of perturbed bits as a part of the whole key. However, with such a remedy, attackers still can use our proposed attack to lessen attack complexity once they get the partial key on the perturbing algorithm. As a result, we have to cancel the relation between different parts of the secret key. A possible solution is to use a single key to simultaneously generate initial conditions/control parameters of the chaotic systems and the secret parameters of the perturbing PRNG, where the total size of generated parameters of the perturbing PRNG should not be smaller than the key size.

§4.6.4 Insulating Digital Chaotic Orbits from Keystream: Yes?

We have known that the possibility to obtain $u_k(t)$ by subtracting perturbing signal from $k(t) = u'_k(t)$ is the practical reason to cause security problems of Hong Zhou et al.'s chaotic ciphers. Then a natural idea to enhance them is to insulate $u'_k(t)$ from $k(t)$ before using it to mask plaintexts, i.e., to make $k(t) = F_{ins}(u'_k(t)) \neq u'_k(t)$, where $F_{ins}(\cdot)$ is a nonlinear insulating function. If the insulating function is carefully selected so that the exact values of $u_k(t)$ can be successfully confused (from an attackers' point of view), then the way to calculate P_i from $k(t)$ will be disabled immediately. There are many methods to do such a task: generating pseudo-random bits from the chaotic orbits by some nonlinear principle[79, 98], combining pseudo orbits of multiple chaotic systems to generate the key-stream[22, 74], pseudo-random permutation of $k(t)$'s bits, transforming pseudo orbits into some pseudo-random patterns[106, 112], etc. Of course, using such a remedy will cause security of the concerned ciphers mainly relies on the insulating algorithm, not the chaotic system itself. Further studies are wanted to estimate whether the above insulating methods can bring other security flaws.

§4.6.5 Avoiding the Use of Weak Keys: Yes

Another simple remedy is to avoid the use of all weak keys. From the strictest viewpoint, only the n-resolution keys are not weak, and the secret key should be uniformly selected at random from V'_n, not S'_n. In such a situation, the key entropy under simple brute force attack decreases from $n-1$ to $n-2$, and the key entropy of our enhanced brute force attack is also $n-2$ since the resolution has been naturally known.

When n is relatively large, we can relax the above rigorous principle, and only avoid the use of "very weak" keys, for example, we can avoid using all keys whose resolutions are less than $n/2$. Such a relaxation can provide a lower bound for the security of related chaotic ciphers against the weak-key based attack.

Disadvantage of this remedy is sacrificing many available keys, so it is a passive countermeasure. But its simplicity in practice makes it an efficient and feasible remedy. Before performances of other remedies are confirmed, we suggest this simple countermeasure.

§4.6.6 Perturbing Chaotic Orbits and also Control Parameters: Yes

In [199], the author suggested perturbing control parameter(s) to improve dynamical degradation of digital chaotic systems. In §3.4.1, we have shown such a method has less better performance than perturbing chaotic orbits. However, here we can combine both two perturbing algorithms to enhance security of Hong Zhou et al.'s ciphers. The perturbation to control parameters is used to confuse the distinguishability of the n probabilities $P_1 \sim P_n$. We think this remedy is also an acceptable solution in practice.

§4.7 Conclusion

This chapter discusses security problems of Hong Zhou et al.'s chaotic ciphers proposed in [24–26] and some possible remedies. The discussions made in this chapter emphasize the following two facts in the design of digital chaotic ciphers: 1) dynamical degradation of digital chaotic systems must be remedied to avoid the degeneration of designed chaotic ciphers, the seeming "perfect" properties of the employed chaotic system in continuum cannot ensure the security in the digital world; 2) even when some method is used to improve the dynamical degradation, there may still exist some black holes to threaten the security. The second fact implies more theoretical studies are wanted on dynamics of digital chaotic systems.

Chapter 5
Cryptanalysis of Searching Based Digital Chaotic Ciphers

§5.1 Introduction

In §2.4.1, we have given a brief description on searching based chaotic ciphers[84, 90, 104, 110, 113–116, 122, 123, 128] and related cryptanalytic work[97, 100, 126]. In this chapter, we will introduce the following works of ours: 1) how to improve E. Alvarez et al.'s chaotic cipher[90] against G. Alvarez et al.'s attacks[97]; 2) our arguments on Jakimoski-Kocarev attack[100] of M. S. Baptista's chaotic cipher[84] (and other ciphers with similar structure[104, 110, 113, 114, 122, 123]); 3) how to enhance M. S. Baptista's chaotic cipher to resist all known attacks[100, 126].

This chapter can be divided into two parts respectively focusing on E. Alvarez et al.'s cipher and M. S. Baptista's cipher. The first part discusses our explanation on why E. Alvarez et al.'s chaotic cipher is so vulnerable to G. Alvarez et al.'s attacks* and an improved scheme against all G. Alvarez et al.'s attacks. In the second part, I show my opinion on practical performance of Jakimoski-Kocarev attack to break M. S. Baptista's cipher and propose a countermeasure to effectively resist Jakimoski-Kocarev attack. Please note that the proposed countermeasure can also be used to resist symbolic dynamics based attacks proposed in [126], since these attacks depend on the same requirement as Jakimoski-Kocarev attack(the occurrence of the number of chaotic iteration).

The organization of this chapter is as follows. In §5.2 we will explain why E. Alvarez et al.'s cipher is not secure by analyzing two essential defects (other non-trivial weaknesses is also discussed). An improved scheme to E. Alvarez et al.'s cipher is proposed and analyzed in detail in §5.3. Both theoretical and experimental analyses show that the improved cipher has satisfactory cryptographic properties (of course, more investigations should be made to confirm its security). In §5.4, we give a detailed introduction on M. S. Baptista's chaotic cipher to facilitate discussion in following sections. Jakimoski-Kocarev attack and our opinion on its performance is given in §5.5. A countermeasure to resist Jakimoski-Kocarev attack and other attacks to M. S. Baptista's cipher is discussed in §5.6. The last section is a summary of this chapter.

*In [97], the authors did not explicitly gave such an explanation on their attacks.

§5.2 E. Alvarez et al.'s Chaotic Cipher and its Essential Defects

§5.2.1 A Brief Introduction

E. Alvarez et al.'s cipher is a symmetric block cipher and encrypts every plain-block into a 3-tuple cipher-block. Different from other conventional block ciphers, its block size is time-variable. Based on a d-dimensional chaotic system $x_{n+1} = F(x_n, x_{n-1}, \cdots, x_{n-d+1})$, the encryption and decryption procedure can be depicted as follows. Firstly, select the control parameter of the system as the secret key, and an integer b_{max} as the maximal block size of plaintext. For one plain-block whose size is $b_i = b_{max}$, choose a threshold U_i to generate a bit chain C_i from the chaotic orbit $\{x_n\}$ according to such a rule: $x_n \leq U_i \rightarrow 0$ and $x_n > U_i \rightarrow 1$. Find the position at which the plain-block appears in C_i and record $(U_i, b_i, \boldsymbol{X}_i)$ as the cipher-block corresponding to the plain-block, where $\boldsymbol{X}_i = (x_i, x_{i-1}, \cdots, x_{i-d+1})$ is the state of the chaotic map at the position. If the plain-block cannot be found in a large enough catalog C_i, $b_i = b_i - 1$ and the search is restarted until the ciphertext can be generated. The tent map (2.5) is used to demonstrate performance of such a chaotic cipher, and the control parameter r is selected as the secret key.

However, only some months later after the proposal of this cipher, G. Alvarez et al. pointed out that it is very easy to be broken when the tent map (2.5) is used [97]. In their paper, they presented four kinds of attacks, which are chosen-ciphertext attack, chosen-plaintext attack, known-plaintext attack and ciphertext-only attack. They also pointed out some other weaknesses of the chaotic encryption system. As a result, the authors claimed that the new chaotic cipher is not secure at all, even if other chaotic systems are used instead of the tent map. In the following subsections, we will investigate two defects in E. Alvarez et al.'s cipher, which are essential reasons to make the cipher insecure.

§5.2.2 Defect 1: The Occurrence of X_i in Ciphertext

The first essential defect lies in the occurrence of $\boldsymbol{X}_i$ in ciphertext. Considering the dynamics of the employed chaotic system depends not only on the secret key (control parameter) but on the initial conditions, an eavesdropper may obtain some useful information from $\boldsymbol{X}_i$ to lessen attack complexity.

Actually, there does exist information leaking in E. Alvarez et al.'s cipher, and the leaking probability $P_l \geq E(1/b_i)$, where $E(x)$ represents the mean value of x. Apparently, since $b_i \leq b_{max}$, we have $P_l \geq E(1/b_i) \geq 1/b_{max}$. Given one cipher-block $(U_i, b_i, \boldsymbol{X}_i)$, let us consider how the b_i bits of the plain-block

$P_i = P_{i,0}P_{i,1}\cdots P_{i,b_i-1}$ is decrypted and how some plain-information is leaked. Since the legal users know the secret key (control parameter), they can calculate the b_i iterating values $\{x_{i+j}\}_{j=0}^{b_i-1}$ from $\boldsymbol{X}_i$. Then the plain-block P_i can be obtained from $\{x_{i+j}\}$ and the threshold U_i as follows:

for $j = 0$ to $b_i - 1$ do
  if $x_{i+j} \leq U_i$ then $P_{i,j} = 0$
  else $P_{i,j} = 1$
end

Obviously, b_0 can be obtained from $\boldsymbol{X}_i$ just by comparing the two values x_i and U_i, without the secret key. Therefore, an illegal user can obtain b_0 in each plain-block under ciphertext-only attack. That is to say, at least $1/b_i$ information of the plain-block leaks from the cipher-block. As a whole, the probability of information leaking P_l will be not less than $E(1/b_i) \geq 1/b_{max}$. Generally speaking, b_{max} cannot be too large, or the encryption speed will be rather slow. Then the information leaking is relatively large to make the chaotic cipher insecure in many serious applications.

Furthermore, if one knows the approximate value r' of the secret key r, he can guess the plain-block by the symbolic dynamics of the chaotic system from the initial condition $\boldsymbol{X}_i$. The closer r' is close to r, the better such a guess works. This fact means that the chaotic cipher is not sensitive to secret key, which is undesired for a good cryptosystem[144, 145]. The authors of [97] employed such a fact to develop a ciphertext-only attack when $r' = 2$. It is found that such a guess can reveal the plain-block with high possibility when the secret key r is close to 2. Of course, the success ratio will decrease as the right key r departs from 2, but bear in mind that the information leaking of this chaotic cryptosystem will not be less than $1/b_{max}$ for all available keys.

In addition, the chosen-ciphertext attack described in [97] is also based on the fact that $\boldsymbol{X}_i$ in ciphertext can expose some useful information about the secret key. By choosing $\boldsymbol{X}_i$ sufficiently close to zero and observing the corresponding plaintext, one can get the secret key in a small number of steps.

§5.2.3 Defect 2: Different Dynamics with Different Keys

For the tent map (2.5), the dynamics with different secret keys (control parameters r) is much different and some dynamical indicators are uniquely determined by the control parameter r. Such dynamical indicators include visited interval of chaotic orbit, Lyapunov exponent, Kolmogrov entropy and the occurrence of periodic window at many control parameters[206, 208, 209, 215]. Since such difference can be extracted from $\boldsymbol{X}_i$ in a number of ciphertexts, it can be used to develop

some available attacks.

The different visited interval of chaotic orbit with different key is easily used to realize chosen-plaintext attack and known-plaintext attack as described in [97]. When control parameter is r, the visited interval will be $[r(1-r/2), r/2]$. By statistics of enough ciphertexts one can get the approximate lower (upper) bound of the visited interval, and then obtain the secret key r approximately. As we have mentioned above, the chaotic encryption system is not sensitive to the secret key, so the approximate secret key is enough to decrypt the ciphertext with high success possibility. Of course, one can find the exact value of the secret key by searching it in a small neighbor area of the approximate value, which will need much less computation complexity than searching it in the whole key space.

Since $\boldsymbol{X}_i$ must be known to make such statistics, the known-plaintext attack and chosen-plaintext attack in [97] depend on the both defects. Hence, if the first defect is avoided, all the four attacks in [97] are infeasible. But in order to avoid other possible attacks in the future, both defects should be cancelled.

§5.2.4 Other Weaknesses

There are also some other weaknesses pointed out by G. Alvarez et al. in [97]. They are the use of too low computing precision, the lack of exact directions about how to choose the initial condition and the secret key, and non-sensitivity of ciphertext to the secret key. The last weakness has been discussed in §5.2.2. Others are not crucial for the original encryption scheme, and can be solved by carefully re-consider implementation of the original cipher.

Besides the above weaknesses, there still exists another serious problem in the original scheme, which is about the slow encryption speed. It is obvious that the encryption speed is chiefly determined by the search for the occurrence of plain-block in C_i. Assume C_i is balanced on $\{0,1\}$, then the probability of the occurrence of every plain-block is $1/2^{b_i}$, so the search procedure can be regarded as Bernoulli experiments with probability $p = 1/2^{b_i}$. The number of experiments satisfies geometric distribution, and its mathematical expectation is 2^{b_i}[212]. If C_i is not balanced, the average number of experiments will be larger than 2^{b_i}. Here, b_i cannot be very small to avoid brute-force attack, and cannot be very large to make the encryption speed much slower than other conventional ciphers. What's more, larger b_i will make the reset probability (i.e., the occurrence probability of $b_i = b_i - 1$ and restarting the search procedure) larger and further make the encryption speed even slower. Such a paradox will make the selection of b_{max} very difficult to obtain both high security and fast encryption speed. In the original cipher, $b_{max} = 16$ is adopted, which makes the chaotic cipher potentially insecure and

makes the encryption speed relatively slow.

§5.3 An Improved Scheme to E. Alvarez et al.'s Cipher

An improved scheme of E. Alvarez et al.'s cipher is proposed in this section. It can avoid the two above-mentioned defects and other weaknesses of the original one, so it can resist all G. Alvarez et al.'s attacks and has better performance. Experimental results confirm cryptographical properties of the improved cipher.

§5.3.1 Description

Without loss of generality, we employ a one-dimensional chaotic map to construct the new cryptosystem. Given a chaotic map defined on the interval $I = [a, b]$ as follows: $x_{n+1} = F(x_n, p)$, where p is the control parameter. The following requirement should be satisfied: the chaotic map is ergodic on I with unique invariant density function[23]. This requirement is needed to avoid the second essential defect, and can make the statistical cryptanalysis impossible. Examples of such chaotic maps are PWLCM-s, such as skew tent map (2.3) and the 1D PWLCM (2.1). Here, please note that the tent map (2.3) is essentially different from the one (2.5) used in the original E. Alvarez et al.'s cipher (although they are both called *tent map*).

Based on a chaotic map satisfying the above requirement, the improved scheme can be described as follows. Please note that it is similar to M. S. Baptista's cipher*, since the iteration number is used as a part of ciphertext.

- *The secret key*: $K = (x_0, p)$, where x_0 is the initial condition of the chaotic map.
- *The input – plaintext*: $P_1 P_2 \cdots P_i \cdots$, where the size of P_i is $b_i \leq b_{max}$.
- *The encryption procedure* is quite similar to the original scheme. For the first plain-block P_1 whose size is $b_1 = b_{max}$, run the chaotic map from x_0, and select a threshold U_1 to generate a bit chain C_1 as the same rule in the original scheme. Find the position at which P_1 appears in C_1, and record (U_1, b_1, n_1) as the cipher-block, where n_1 is the number of iterations of the chaotic map from x_0. If P_1 cannot be found in a large enough catalog C_1, $b_1 = b_1 - 1$ and the search restarts. For the second and the following plain-blocks, the encryption procedure is just like above, except that the chaotic map runs from the position after the last plain-block is encrypted, not x_0.

*Although the improved cipher can be considered as a combination of E. Alvarez et al.'s cipher and M. S. Baptista's cipher, actually the similar idea was independently re-proposed by us. When I submitted my paper [110] in July 2001, I have not got a copy of [84] and read it.

- *The output – ciphertext*: $(U_1, b_1, n_1)\ (U_2, b_2, n_2) \cdots (U_i, b_i, n_i) \cdots$. In fact, the threshold U_i can be fixed for all plain-blocks, then the ciphertext will be simplified to $(b_1, n_1)(b_2, n_2) \cdots (b_i, n_i) \cdots$. Generally speaking, the threshold should be selected to make C_i balanced on $\{0, 1\}$, i.e., $P\{C_i = 0\} = P\{C_i = 1\}$. It can be derived from the invariant density function of the chaotic map. For the two chaotic maps above-mentioned, the threshold is 0.5, i.e., the midpoint of $I = [0, 1]$.
- For the legal users knowing the secret key, *the decryption procedure* is easy to be realized by re-generating C_i for each cipher-block.

We can see both defects existing in the original cipher are avoided in the above scheme. What's more, different from the original cipher, the improved one is more of a stream cipher than a block cipher. Such a fact means that smaller b_{max} can be used and the paradox between the security and the encryption speed will be relaxed to some extent.

As we suggested in §2.5.2, dynamical degradation of digital chaotic systems should be remedied with pseudo-random perturbation. It is obvious that the perturbation should be small enough to avoid destroying the essential dynamics of original digital chaotic systems. However, for the above cipher, because the chaotic system is used to generate unpredictable pseudo-random bit sequence with balanced property and long cycle, so that the exact dynamics itself is not important. As a result, the perturbing signal can be relatively large. In fact, our experiments have shown that: the larger the perturbing signal is, the better results coincide with the theoretical analyses and the faster search procedure finishes (i.e., the faster the encryption system runs). It is because distribution of the chaotic orbit will become more uniformly with larger perturbing signal and the generated bit sequence will be more ideal. So we suggest using larger perturbing signal instead of smaller one, then the perturbed chaotic system can be considered as a hyper-complex nonlinear system composed of digital chaos and pseudo-randomness of the perturbing PRNG. Here, nonlinearity of the chaotic systems ensures the security, and the perturbing PRNG ensures the ideal property of generated bit sequence by flattening decayed digital chaos.

§5.3.2 Cryptographic Properties

In this subsection, let us give the following statement firstly. Since b_i in ciphertext just indicates the block size of the corresponding plain-block, we will only regard n_i as the "real" ciphertext.

As we know, two chief cryptographic properties of a good cipher are confusion and diffusion, which are commonly ensured by the balance and avalanche

properties of the ciphertext in conventional cryptography [144, 145]. But the above improved cipher has rather different property: the ciphertext is not balanced, since the larger n_i, the smaller the possibility of its occurrence in the ciphertext. Assume C_i is a balanced i.i.d. (independent with identical distribution) bit sequence, the search procedure can be considered as Bernoulli experiments with probability $1/2^{b_i}$; then we can deduce the discrete probability distribution of n_i:

$$P\{n_i = k\} = \frac{1}{2^{b_i}} \cdot \left(1 - \frac{1}{2^{b_i}}\right)^{k-1}, \tag{5.1}$$

which is independent and identical for different secret keys and plaintexts theoretically.

Then how the proposed cipher exhibits confusion and diffusion with unbalanced probability? Actually, there are four corresponding facts on the above distribution of the ciphertext: 1) for different plaintexts, the ciphertext has the same distribution function; 2) for different secret keys (control parameters and initial conditions), the ciphertext has the same distribution function; 3) for two plaintexts even with only one bit difference, the ciphertext is rather different; 4) for two secret keys even with only one bit difference, the ciphertext is rather different. The first two facts denote confusion, and the other two denote diffusion.

Because our improved scheme avoids the two essential defects in the original E. Alvarez et al.'s cipher, and satisfies the confusion and diffusion properties, we can use smaller b_{max} compared to the original scheme. Thus the encryption speed will be faster. However, since the time-consuming search procedure is still used, the encryption speed is still slower than most conventional ciphers. Assume the speed of iterating the chaotic map is s iterations per second; the average encryption speed will be $s \cdot E(b_i)/E(n_i) \approx s \cdot b_{max}/2^{b_{max}}$ bps (bits per second). Hence, such a chaotic encryption scheme only can be used in non-real-time applications, such as the secure transmission of short messages over network or the secure storage of small files in computers. Of course, when $b_{max} = 1, 2$, the encryption speed will be rather fast: about $s/2$ bps. However, further studies are needed to find potential negative influence of using so small b_{max}.

At last, let us discuss the key entropy of the improved scheme. When the finite computing precision is n(bits), the control parameter and initial conditions are represented as a fixed-point binary decimal. So the key entropy of the improved scheme is $2n$. In digital computers n can be selected as 32 or 64, then the key entropy is 64 or 128, which is enough as a secure cipher. If higher security is wanted, larger n is suggested.

§5.3.3 Compression after Encryption

There is a nontrivial problem in the chaotic cipher: the size of ciphertext is (> 2 times when $n_i \leq 2^{bmax+1}$) larger than the size of plaintext. Compressing ciphertext can solve it. Since the ciphertext is not balanced, it can be consequently compressed with lossless statistical compression algorithm, such as Huffman coding algorithm[216]. Firstly, divide the ciphertext into two bit streams: $b_1 b_2 \cdots b_i \cdots$ and $n_1 n_2 \cdots n_i \cdots$. Then compress the two sub-streams with Huffman coding separately. For the stream $n_1 n_2 \cdots n_i \cdots$, the average size of compressed cipher-block will be b_{max} according to Eq. (5.1). For the stream $b_1 b_2 \cdots b_i \cdots$, the average size of compressed cipher-block will be close to 1 since each b_i in ciphertext trends to be b_{max}. Hence, the average size of compressed cipher-block will be close to $b_{max} + 1$, only 1 bit more than the maximal size of plain-block.

§5.3.4 Experimental Results

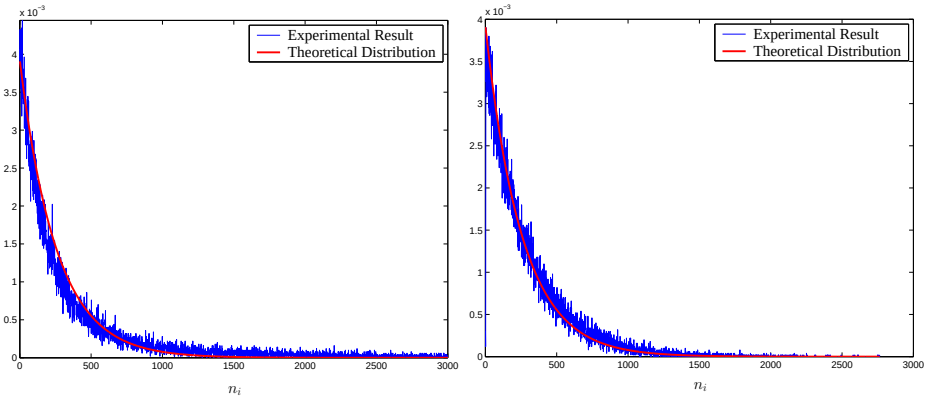
Based on the tent map (2.3), we constructed an experimental cipher and test its cryptographic properties. Here the computing precision is $n = 32$(bits), $b_{max} = 8$, and the perturbing PRNG is a maximal length LFSR (m-LFSR), whose order is n and primitive polynomial is $1 + x + x^{27} + x^{28} + x^{32}$[214]. The perturbing interval $\Delta = n^*$, and all bits of the m-LFSR are used to perturb the chaotic orbit.

As we mentioned above, the discrete distribution of n_i is denoted by Eq. (5.1) theoretically. When the plaintext is distributed uniformly, the experimental result coincides with the theoretical curve as shown in Figure 5.1a. When the plaintext is $59\ 59 \cdots 59 \cdots$, the experimental result is shown in Figure 5.1b. The number of encrypted plain-block is 50,000. When the secret keys (control parameters and initial conditions) are selected as different values randomly, similar results are obtained. So the confusion property is confirmed.

Other experiments were made to verify the diffusion property. The difference of n_i in two ciphertexts is shown in Figure 5.2a–c, with the least difference respectively in plain-texts, control parameters and initial conditions. The different parameters are listed as follows:

- The least different plaintexts (see Figure 5.2a): **195** $195 \cdots 195 \cdots$ and **196** $195 \cdots 195 \cdots$;
- The least different control parameters (see Figure 5.2b): $p_1 = \mathbf{31849}/2^{32}$ and $p_2 = \mathbf{31848}/2^{32}$;

*Please note $\gcd(\Delta, 2^n - 1) = 1$, which is helpful to obtain the maximal cycle length of the bit sequence C_i.



a) The distribution of n_i with uniformly distributed plaintext

b) The distribution of n_i with fixed plaintext 59 59 $\cdots$ 59 $\cdots$

Figure 5.1: The discrete probability distribution of n_i

- The least different initial conditions (see Figure 5.2c): $x_{0,1} = \mathbf{40332}/2^{32}$ and $x_{0,2} = \mathbf{40333}/2^{32}$.

§5.4 M. S. Baptista's Chaotic Cipher and its Modified Versions

In this section, let us give a detailed introduction of M. S. Baptista's chaotic cipher and its modified versions to make the following description on Jakimoski-Kocarev attack clearer (with a rather different way from the one in [84]). Given a one-dimensional chaotic map $F : X \to X$, divide an interval $[x_{min}, x_{max}) \subseteq X$ into S ϵ-intervals $X_1 \sim X_S$: $X_i = [x_{min} + (i-1)\varepsilon, x_{min} + i\varepsilon)$, where $\varepsilon = (x_{max} - x_{min})/S$. Assume plain messages are composed by S different characters $\alpha_1, \alpha_2, \cdots, \alpha_S$, use a bijective map

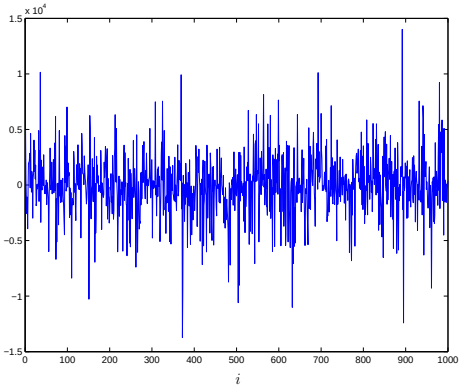
$$f_S : \boldsymbol{X}_\epsilon = \{X_1, X_2, \cdots, X_S\} \to \boldsymbol{A} = \{\alpha_1, \alpha_2, \cdots, \alpha_S\} \tag{5.2}$$

to associate the S different ϵ-intervals with the S different characters. Define a new function $f'_S : X \to \boldsymbol{A}$: $f'_S(x) = f_S(X_i)$, if $x \in X_i$.

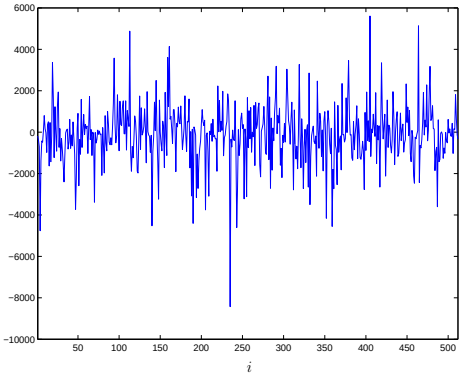
Given a plain-message $M = \{m_1, m_2, \cdots, m_i, \cdots\}$ ($m_i \in \boldsymbol{A}$), M. S. Baptista's cipher can be described as follows.

- *The chaotic system*: Logistic map $F(x) = rx(1-x)$.
- *The secret key*: the association map S^*, the initial condition x_0 and the control

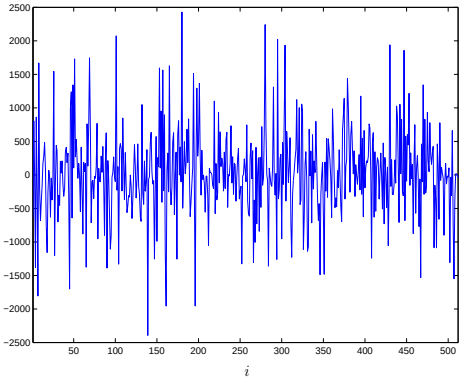
*We think that the map f_S should not be included in the secret key from implementation consideration and Kerckhoffs' principle [144, 145].



a) The difference of n_i with two least different plaintexts



b) The difference of n_i with two control parameters (2^{-n} difference)



c) The difference of n_i with two initial conditions (2^{-n} difference)

Figure 5.2: The difference of n_i in two ciphertexts

parameter r of Logistic system.

- *Encryption*:
  - *For the first plain-character* m_1: Iterate the chaotic system from x_0 to find a chaotic state x that satisfies $f'_S(x) = m_1$, and record the iteration number C_1 as the first cipher-message unit and $x_0^{(1)} = F^{C_1}(x_0)$;
  - *For the* i^{th} *plain-character* m_i: Iterate the chaotic system from $x_0^{(i-1)} = F^{C_1+C_2+\cdots+C_{i-1}}(x_0)$ to find a chaotic state x satisfying $f'_S(x) = m_i$, record the iteration number C_i as the i^{th} cipher-message unit and $x_0^{(i)} = F^{C_i}\left(x_0^{(i-1)}\right)$.
- *Decryption*: For each ciphertext unit C_i, iterate the chaotic system for C_i times from the last chaotic state $x_0^{(i-1)} = F^{C_1+C_2+\cdots+C_{i-1}}(x_0)$, and then use $x_0^{(i)} =$

$F^{C_i}\left(x_0^{(i-1)}\right)$ to derive the plain-character m_i by the association map f_S.

- *Constraints of C_i*: Each cipher-message unit C_i should yield to the constraint $N_0 \leq C_1 \leq N_{max}$ ($N_0 = 250$ and $N_{max} = 65532$ in [84]). Since there exist many options for each C_i in $[N_0, N_{max}]$, an extra coefficient $\eta \in [0, 1]$ is used to choose a right number: if $\eta = 0$, C_i is chosen as the minimal number satisfying $f'_S(x) = m_i$; if $\eta \neq 0$, C_i is chosen as the minimal number satisfying $f'_S(x) = m_i$ and $\kappa \geq \eta$ simultaneously, where κ is a pseudo-random number with normal distribution within the interval $[0, 1]$.

- *A captious note from me*: In [84], unlike E. Alvarez et al.'s cipher, M. S. Baptista did not say what we should do once $C_i > N_{max}$. It seems that M. S. Baptista think C_i will never be greater than N_{max}. However, from the strictest point of view, I do not think so. Here, assume $F(x)$ visits each ϵ-interval with uniform probability $p = 1/S$, we can deduce

$$P\{C_i > N_{max}\} = P\{C_i - N_0 > N_{max} - N_0\} = (1-p)^{N_{max}-N_0}. \tag{5.3}$$

  See Figure 5.3 for a view of the above probability versus $N_{max} - N_0$. We can see when $N_{max} - N_0 > 10000$ this probability goes to zero in IEEE double precision floating-point arithmetic[217]. Although this probability is very small when N_{max} is large enough, it is not yet equal to zero. To make the cipher rigorous, we can add such a rule into the above encryption/decryption procedure: when $C_i = N_{max}$, the ciphertext is a 2-tuple data (N_{max}, m_i). Since $P\{C_i > N_{max}\}$ is small enough, such a tiny information leaking will not lower security at all. Of course, if N_{max} is selected to be large enough, we think it is reasonable to discard this captious note. In the following contexts, I will assume N_{max} is always large enough and assume $P\{C_i > N_{max}\} = 0$ (this assumption will be used in §5.6).

The above chaotic cipher has two defects: a) the distribution of the ciphertext is not balanced, and the occurrence probability decays exponentially as C_i increases from N_0 to N_{max}; b) at least N_0 chaotic iterations are needed to encrypt a plain-character, which makes the encryption speed very slow compared with other conventional ciphers.

In [104], W.-K. Wong et al. improved the above original cryptosystem as follows: for each plain-character m_i, firstly generate a pseudo-random number r_C distributed uniformly between 0 and a pre-defined maximum r_{max}, iterate the chaotic system for r_C times and then iterate it until a chaotic state x satisfying $f'_S(x) = m_i$ is found, record the iteration number as the cipher-message unit C_i. Such a modified cryptosystem can avoid the first defect of the original one, but

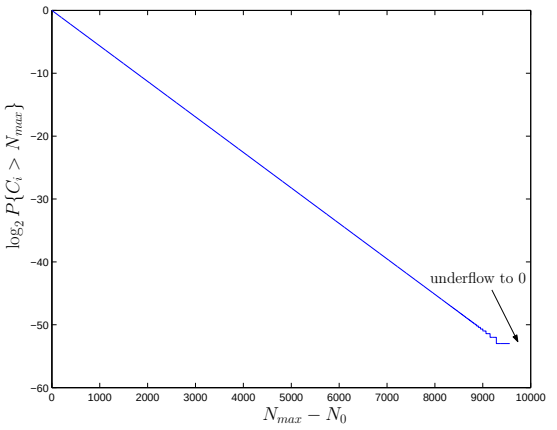


Figure 5.3: $P\{C_i > N_{max}\}$ versus $N_{max} - N_0$

makes the second defect even worse (averagely much more chaotic iterations are needed to cause much slower encryption speed). In [114], K.-W. Wong suggested dynamically updating the association map f_S (called look-up-table in Wong's paper) to promote the encryption speed. But a number of chaotic iterations are still needed so the encryption is still much lower than conventional ciphers. A very recent progress was made again by K.-W. Wong et al. in [123], in which a session key is introduced to realize synchronization between sender and receiver and encryption/decryption starts after a synchronization period with multiple iterations. The main contribution of the recent paper is to make cipheretext shorter by replacing C_i with index of each plaintext block in the dynamic look-up-table.

§5.5 Jakimoski-Kocarev Attack and its Performance

§5.5.1 A Introduction to Jakimoski-Kocarev Attack

In [100], G. Jakimoski and L. Kocarev proposed a known-plaintext attack to break the original M. S. Baptista's chaotic cipher. The cryptanalysis is based on the following fact: one can establish an association table between the moment of interest and the plain-character by observing a number of plaintext/ciphertext pairs, where "the moment of interest" of a ciphertext unit C_i is $n = \sum_{j=1}^{i} C_j$, i.e. the *total* number of chaotic iterations from x_0 to the current chaotic state. The association table can be used to decrypt the corresponding plain-character if the same moment n re-occurs in the ciphertext stream. In [100], an example is given to explain this attack: assume "subject" and "to" are two known plaintexts and they are encrypted as *272 258 305 285 314 276 422* and *254 267* respectively. Then one

can obtain an association table shown in Table 5.1. Using the constructed table, he can decrypt any ciphertext that corresponds to a moment of interest listed in this table. For example, a ciphertext *272 249* can be immediately decrypted as "so" (*272* denotes "s" and $272 + 249 = 521$ denotes "o"). Apparently, if more plaintext/ciphertext pairs are known, this table will contain more associations, and then more ciphertexts can be decrypted by this table.

Table 5.1: An association table constructed from two known plaintexts "subject" and "to"

n	*254*	*272*	*521*	*530*	*835*	*1120*	*1434*	*1710*	*2132*
m_i	t	s	o	u	b	j	e	c	t

Apparently, Jakimoski-Kocarev attack mainly depends on the existence of C_i in the ciphertext, so it is possible to use it to break all modifications of M. S. Baptista's cipher and the improved E. Alvarez et al.'s cipher we proposed in §5.3 and [110]. In the following contexts, we will chiefly pay our attention on the original M. S. Baptista's cipher.

§5.5.2 My Argument on Performance of Jakimoski-Kocarev Attack

In [100], the authors stated that "*Statistical tests show that over 90% of the moments of interest can be recovered using only 4000 plaintext/ciphertext pairs*". It seems that this attack is rather perfect as a tool to break related chaotic cryptosystems. However, in my opinion, its performance is not so effective as the proposers claimed. The following are some reasons to support my argument.

Fact 1: *to decrypt one ciphertext unit, averagely more than one plain-characters should be known.* If an attacker gets to know a plaintext with i different characters, he can construct a table with i different associations, and then he can use the i associations to decrypt i ciphertext units. That is to say, to decrypt one ciphertext unit, one plain-character must be known firstly. When the number of known plain-characters N_p increases, the number of decrypted ciphertext units (i.e., the moments of interest) N_c will also increase. However, the increment ratio of N_c will be less than the ratio of N_p, since plain-characters in different plaintexts may generate the same associations in the table. As the number of plaintexts increases, the ratio of N_c will become even less and less. See Figure 5.4 for an experimental curve about N_c versus N_p. Consequently, to decrypt one ciphertext unit, averagely more than one plain-characters are required. Apparently, Jakimoski-Kocarev attack works like an exhaustive attack.

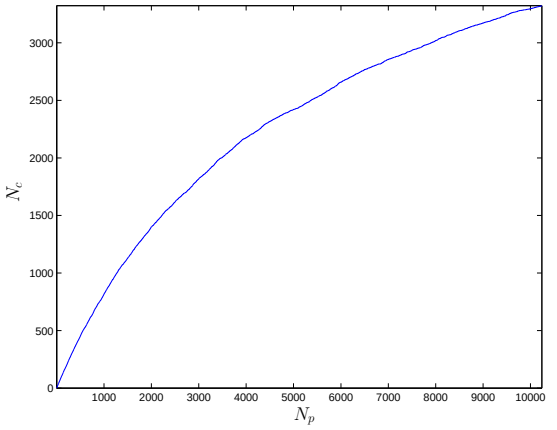


Figure 5.4: N_c versus N_p
(Related parameters are $S = 256$, $b = 3.78$, $x_0 = 0.43203125$, $x_{min} = 0.2$, $x_{max} = 0.8$. 1024 plaintexts with 10 random plain-characters are used.)

Fact 2: *if all known plaintexts contain at most i plain-characters, it is almost impossible to decrypt any plain-character whose position is far beyond i and absolutely impossible to decrypt any plain-character whose moment of interest is beyond* $i \cdot N_{max}$. For a given plaintext, if the first i plain-characters (and the corresponding ciphertext units) are known, it is absolutely impossible to decrypt any following plain-character in this plaintext. What's more, even if the first i plain-characters of a lot of plaintexts are known, it is probabilistically impossible to decrypt any plain-character whose position is far beyond i. In fact, because of the exponentially decayed occurrence probability of C_i (please see Fig. 3 of [84] and Fig. 1 of [104]), the probability of successful attack will decrease exponentially as the position of plain-character goes away from i and decrease to zero once the moment of interest becomes larger than $i \cdot N_{max}$. For example, given a plaintext "*Can you give me any help to break this chaotic cryptosystem*" encrypted by the original M. S. Baptista's cipher, assume an attacker can only get to know plaintexts with 3 plain-characters, it will be almost (probabilistically) impossible for him to decrypt the last word "*cryptosystem*" although he can decrypt the first word "*Can*" with rather high probability.

Fact 3: *M. S. Baptista's cipher is more of a stream cipher than a block cipher, since same plain-characters may be encrypted as different ciphertext units. But Jakimoski-Kocarev attack is designed following the idea of breaking block ciphers, which is not suitable for stream ciphers.* Consider a general XOR-based stream cipher with the secret key-stream $\{k_i\}$, there exists a similar known-plaintext attack to Jakimoski-Kocarev's: once the first l plain-characters of one plaintext are known by an at-

tacker, he can XOR the plain-characters and corresponding cipher-characters to derive the first l keys $k_1 \sim k_l$, and then the first l plain-characters of any plaintext encrypted with a same key-stream can be decrypted successfully (but all following plain-characters still remain secure). Generally speaking, such an attack cannot be considered as a practical attack from strict cryptographic viewpoint, since it cannot break the secret key generating the key-stream $\{k_i\}$ and cannot reveal the following plain-characters by previous known ones[144, 145]. Similarly, Jakimoski-Kocarev attack is not a strong tool to break related chaotic cryptosystems, either.

Fact 4: *it will be impossible to decrypt any plain-character in a plain-message, if not all previous $i-1$ units of the ciphertext are known.* To calculate the moment of interest of a ciphertext unit C_i, all $i-1$ previous ciphertext units $C_1 \sim C_{i-1}$ must be known: $n = \sum_{j=1}^{i} C_j$. As a natural result, it will be absolutely impossible for an attacker to decrypt even one plain-character if he does not observe and record all previous ciphertext units. For example, given a plaintext "Who am I", if an attacker only observes the ciphertext units of "ho am I", he cannot get any association to decrypt other ciphertexts. This fact lowers the practical applicability of Jakimoski-Kocarev attack.

From the above facts, we can see that Jakimoski-Kocarev attack is not so effective as the authors argued in [100]. But how to understand the statement "*... over 90% of the moments of interest can be recovered using only 4000 plaintext/ciphertext pairs*"? Assume the maximal length of plaintexts is l_{max}, the maximal value of moments of interest will be $(N_{max} - N_0 + 1) \cdot l_{max}$. From Fact 1 and 2, the number of moments of interest N_c that can be obtained from 4000 known plaintexts will satisfy $N_c < N_p \cdot l_{max} = 4000 \cdot l_{max}$, which is much smaller than $90\% \cdot (N_{max} - N_0 + 1) \cdot l_{max} = 0.9 \cdot (65532 - 250 + 1) \cdot l_{max} = 58754.7 \cdot l_{max}$. Apparently, the statement of "90%" is ambiguous and inadequate. In fact, it is conceptually right that 90% of S values of plain-characters can be obtained in the association table by 4000 plaintext/ciphertext pairs. But such a fact cannot be used to show the effectiveness of the attack at all, because different ciphertext units may correspond to the same plain-characters in M. S. Baptista's cipher (recall Fact 3).

§5.6 A Remedy to Resist Jakimoski-Kocarev Attack

Although Jakimoski-Kocarev attack is not very effective to break related chaotic ciphers, it is still useful in some situations to lessen attack complexity. In this section, we will present a simple remedy to provide satisfactory resistance to Jakimoski-Kocarev attack. Such a remedy is available for all related cryptosys-

tems [84, 104, 110]. Also, our proposal can also effectively resist new attacks proposed in [126], since these new attacks rely on the same fact as Jakimoski-Kocarev attack: the occurrence of C_i in the ciphertext.

§5.6.1 Description

Before explaining the remedy, let us see why Jakimoski-Kocarev attack works. As we know, each ciphertext unit C_i is the iteration number for the chaotic system (from $x_0^{(i-1)}$) to reach the ϵ-interval representing the current plain-character m_i, then $C_1 \sim C_i$ can be accumulated together to recover the moment of interest $n = \sum_{j=1}^{i} C_j$. If the plain-character m_i is known by an attacker, he can directly get the association between the moment of interest n and the plain-character m_i, and use this association to decrypt any ciphertext unit that corresponds to the same moment of interest n.

Apparently, if we cut off the way to construct the associations between the moments of interest and the plain-characters, Jakimoski-Kocarev attack will be disabled immediately. Here, we will employ chaotic masking algorithm to realize such a task. Chaotic masking algorithm is somewhat like "whitening" technique used in DES^X, Khufu and Khafre cryptosystems[144, §15.6].

A natural idea to frustrate Jakimoski-Kocarev attack is to cut off the way of an attacker to calculate the value of $n = \sum_{j=1}^{i} C_j$. How can we do so? A simple answer is to mask the ciphertext C_i with the current chaotic state $x_0^{(i)} = F^{C_1+C_2+\cdots+C_i}(x_0)$. Since C_i is a 16-bit number ($250 \leq C_i \leq 65532$) and generally $x_0(i)$ has more bits, some bit-extracting function should be used to select 16 bits from the binary representation of $x_0^{(i)}$ to mask C_i. Please note that the bit extracting function cannot be freely selected to avoid information leaking of the current chaotic state, which will be discussed in the next subsection. The masking operation can be any nonlinear function, such as XOR or modular addition.

Assume the bit-extracting function is $f_{be}(\cdot)$ and the masking operation is $\oplus$, we can use the remedy to enhance the original M. S. Baptista's cipher (and related chaotic ciphers[104, 110, 113, 114, 122, 123]) as follows.

Encryption. For the i^{th} plain-character m_i, iterate the chaotic system from $x_0^{(i-1)}$ to find a suitable chaotic state x satisfying $f'_S(x) = m_i$ (and other requirements defined by N_0, N_{max}, η, κ), record the number of chaotic iterations from $x_0^{(i-1)}$ as $\widetilde{C}_i$ and $x_0^{(i)} = F^{\widetilde{C}_i}(x_0^{(i-1)})$. Then the i^{th} cipher-message unit of m_i is $C_i = \widetilde{C}_i \oplus f_{be}\left(x_0^{(i)}\right)$.

Decryption. For each ciphertext unit C_i, firstly iterate the chaotic system for N_0 times and set $\widetilde{C}_i = N_0$, then do the following operations (if $\eta \neq 0$, such operations can be made only when $\kappa \geq \eta$): if $\widetilde{C}_i \oplus f_{be}(x) = C_i$ then use the

current chaotic state x to derive the plain-character m_i and goto the next ciphertext unit C_{i+1}; otherwise iterate the chaotic system once and $\widetilde{C}_i++$, until the above condition is satisfied.

In the following subsection, we will show that cryptographically strong $f_{be}(\cdot)$ can conceal the exact value of C_i and make Jakimoski-Kocarev attack and symbolic dynamics based attacks in [126] impossible. However, carefully investigate decryption procedure above, we can see wrong plain-characters may be "decrypted" with a small probability: when $\widetilde{C}_i \oplus f_{be}(x) = C_i$, the restored "$C_i$" may be a pseudo-value of the real C_i at the encryption side so that the restored chaotic state x is wrong. Unfortunately, in our paper [128], we casually neglected this problem. Now we will try to mend the above encryption/decryption scheme to solve this defect. I will submit a note to *Physics Letters A* soon to rectify our works reported in [128].

Rectifying the above Remedy Proposed in [128]

At first, to see how serious this problem is, let us estimate the value of the error probability at the encryption side as follows. When and only when the real C_i never occurs before x satisfying $f'_S(x) = m'$ (and other requirements defined by N_0, N_{max}, η, κ) is found for the first time, the decryption will be correct. That is to say, for a specific $\widetilde{C}_i$, the probability to get successfully restore $\widetilde{C}_i$ (i.e. the probability to get a right decipher) via the above decryption procedure is

$$\begin{aligned} P_c\left(\widetilde{C}_i\right) &= P\left\{\bigwedge_{k=N_0}^{\widetilde{C}_i-1}\left(k \oplus f_{be}\left(F^k\left(x_0^{(i-1)}\right)\right) \neq C_i\right)\right\} \\ &= P\left\{\bigwedge_{k=N_0}^{\widetilde{C}_i-1}\left(f_{be}\left(F^k\left(x_0^{(i-1)}\right)\right) \neq k \oplus C_i\right)\right\}. \end{aligned} \tag{5.4}$$

Generally, assume the bit size of C_i is n (for M. S. Baptista's cipher, $n = 16$) and the chaotic orbit $\left\{F^k\left(x_0^{(i-1)}\right)\right\}$ has uniform distribution, $\forall C_i, P\left\{f_{be}\left(F^k\left(x_0^{(i-1)}\right)\right) = C_i\right\} = 2^{-n}$, i.e. $P\left\{f_{be}\left(F^k\left(x_0^{(i-1)}\right)\right) \neq k \oplus C_i\right\} = 1 - 2^{-n}$. Assume $f_{be}\left(F^k\left(x_0^{(i-1)}\right)\right) = k \oplus C_i (k = N_0 \sim \widetilde{C}_i - 1)$ are independent events, we can deduce $P_c\left(\widetilde{C}_i\right) = (1-2^{-n})^{\widetilde{C}_i - N_0}$. It is obvious that $P_c\left(\widetilde{C}_i\right) \to 0$ as $\widetilde{C}_i \to \infty$, which means the decryption behaves like random guess after a enough long time.

Considering the non-uniform distribution of $\widetilde{C}_i$, for the first plain-character m_1, from the total probability rule the final probability $P_{c,1}$ can be calculated to

be*:

$$P_{c,1} = \sum_{k=N_0}^{N_{max}} P\{\widetilde{C}_i = k\} \cdot P_c(k) = \sum_{k=N_0}^{N_{max}} P\{\widetilde{C}_i = k\} \cdot (1-2^{-n})^{k-N_0}. \tag{5.5}$$

To simplify calculation without loss of generality, assume $F(x)$ visits each ϵ-interval with the same probability $p = 1/S$†, we have $P\{\widetilde{C}_i = k\} = p(1-p)^{k-N_0}$, then we can get

$$\begin{aligned} P_{c,1} &= \sum_{k=N_0}^{N_{max}} p(1-p)^{k-N_0} \cdot (1-2^{-n})^{k-N_0} \\ &= \sum_{k'=0}^{N_{max}-N_0} p \cdot q^{k'} = p \cdot \frac{1-q^{N_{max}-N_0}}{1-q}, \end{aligned} \tag{5.6}$$

where $q = (1-p) \cdot (1-2^{-n})$. When $S = 256, n = 16, N_0 = 250, N_{max} = 65532$ (original values in M. S. Baptista's cipher), $P_c \approx 0.9961240899211138$ (calculated in MathWorks' Matlab® with IEEE double precision floating-point arithmetic). Considering $1/(1-P_{c,1}) \approx 258$, we can see one plaintext with wrong leading plain-character will occur averagely in 258 plaintexts. Here please note that all plain-bytes after a wrong plain-character will be wrong with a high probability close to 1, i.e., there exists error propagation. It is obvious that the error propagation makes things worse for $i > 1$:

$$P_{c,i} = \left(\prod_{j=1}^{i-1} P_{c,j}\right) \cdot \frac{p\left(1-q^{N_{max}-N_0}\right)}{1-q} = \left(\prod_{j=1}^{i-1} P_{c,j}\right) \cdot P_{c,1} = P_{c,1}^{i}. \tag{5.7}$$

As the increment of i, the probability decreases exponentially. Once $P_{c,i}$ goes below $1/S$, then random guess will replace the role of the problematic decipher.

When the encrypted file is a article or an digital image, the wrong plain-bytes may be easily distinguished by humans. Apparently, such a feature will be useful to realize a novel and interesting cipher similar to visual cryptography[218]. In future we will study whether or not it is possible to extend this *probabilistic decryption* cipher to conventional cryptography.

Now let us return to the main thread, since $P_{c,i} < 1$ we have to modify our remedy to make $P_{c,i} \equiv 1$. To do so, we change the encryption/decryption procedure as follows:

Encryption. A memory unit is allocated to store $N_{max} - N_0 + 1$ variables $B[N_0] \sim B[N_{max}]$ representing $C_i = N_0 \sim C_i = N_{max}$. For the i^{th} plain-character

*Assume $P\{C_i > N_{max}\} = 0$, see my captious note in §5.4 for explanation.

†Logistic map does not rigorously satisfy this requirement, so we suggest using PWLCM-s to replace Logistic map in original M. S. Baptista's cipher.

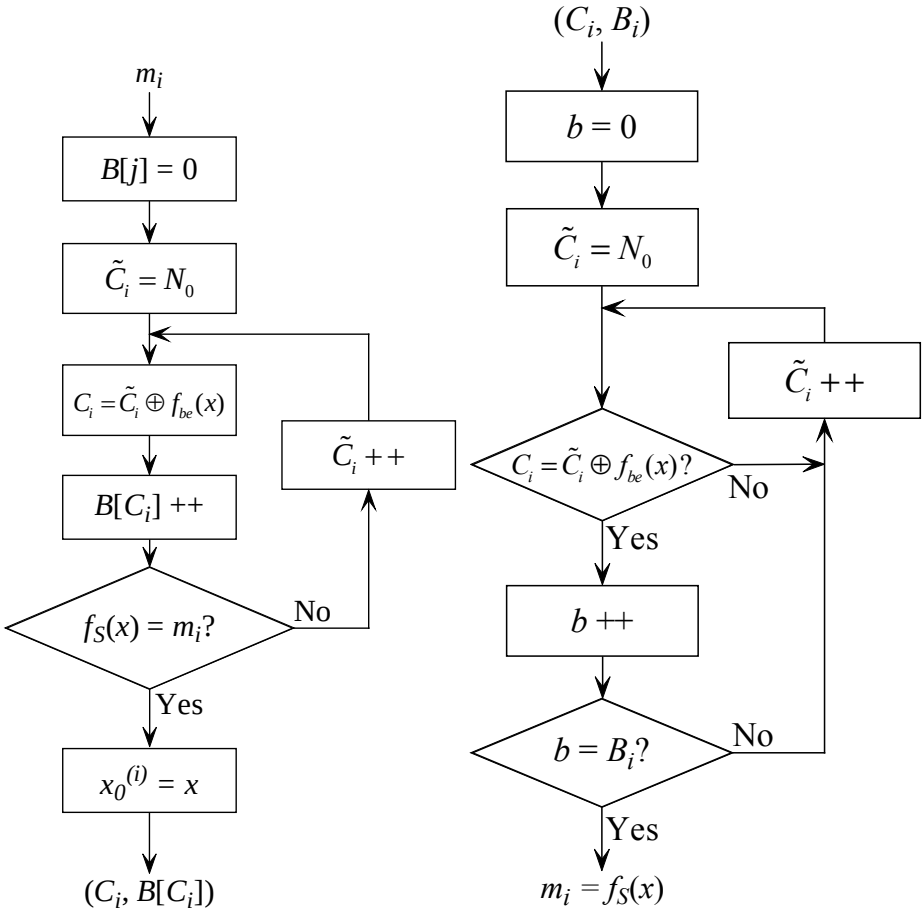
m_i, firstly reset all $B[j](i = N_0 \sim N_{max})$ to zeros, and iterate the chaotic system from $x_0^{(i-1)}$ for N_0 times, set $\widetilde{C}_i = N_0$, and then do the following operations: $C_i = \widetilde{C}_i \oplus f_{be}(x)$, $B[C_i] + +$, if the current chaotic state x satisfying $f_S(x) = m_i$, then a 2-tuple ciphertext is generated $(C_i, B[C_i])$ and set $x_0^{(i)} = x$ and goto the next plain-character m_{i+1}, otherwise repeat this procedure until a ciphertext is generated.

Decryption. For each ciphertext unit (C_i, B_i), firstly iterate the chaotic system for N_0 times and set $\widetilde{C}_i = N_0$, then do the following operations (if $\eta \neq 0$, such operations can be made only when $\kappa \geq \eta$): if $\widetilde{C}_i \oplus f_{be}(x) = C_i$ for the B_i^{th} times then use the current chaotic state x to derive the plain-character m_i and goto the next ciphertext unit (C_{i+1}, B_{i+1}); otherwise iterate the chaotic system and $\widetilde{C}_i + +$ for once iteration, until the above condition is satisfied.

In Figure 5.5 we give flow charts of the above modified encryption and decryption procedure, in which $\boxed{B[j] = 0}$ denotes resetting all $B[j]$ to zeros, $\boxed{\widetilde{C}'_i = N_0}$ denotes N_0 chaotic iterations plus $\widetilde{C}'_i = N_0$, and $\boxed{\widetilde{C}'_i + +}$ denotes one chaotic iteration plus $\widetilde{C}'_i + +$.

Compared with the original M. S. Baptista's cipher, the above modification enhances security against related attacks at the cost of adding implementation complexity: 1) the encryption speed will become lower since $N_{max} - N_0 + 1$ variables $B[j]$ should be set to zeros for each plain-character; 2) the size of the ciphertext is expanded even more ($B[C_i]$ is added into each ciphertext unit); 3) an extra memory unit is needed to store $N_{max} - N_0 + 1$ variables $B[j]$, when $B[j]$ is stored as a 2-byte integer, the memory size is $2 \times (N_{max} - N_0 + 1)$ bytes (when $N_{max} = 65532$ and $N_0 = 250$, it is not greater than 128 KB). Fortunately, the requirement on extra memory is acceptable in almost all digital computers (actually 128 KB is not so much for a computer with over 100 MB memory), and the encryption speed will be not influenced much when the above cipher is realized in hardware supporting parallel computation*. Therefore, we think the above modification is still useful in actual applications. Of course it will be better if the encryption speed of the cipher can be promoted and the ciphertext size can be smaller (K.-W. Wong's proposal in [123] can be considered as a possible candidate).

*All $N_{max} - N_0 + 1$ variables $B[j]$ can be reset to zeros within a clock cycle simultaneously, which cancels the main negative factor to influence the encryption speed. In addition, chaotic iteration can be run parallel with $C_i = \widetilde{C}_i \oplus f_{be}(x)$, $B[C_i] + +$ and $f_S(x) = m_i$? with pre-calculation and delay design.



a) Encryption procedure b) Decryption procedure

Figure 5.5: Decryption/decryption procedure of our modified M. S. Baptista's cipher

§5.6.2 Discussion

The above modified cipher seems to be immune to Jakimoski-Kocarev attack, since it is impossible for an attacker to calculate $\sum_{j=1}^{i} C_j$ only by observing plaintext/ciphertext pairs.

However, we should carefully configure the modified cipher to avoid a new insecurity problem induced by the bit extracting function $f_{be}(\cdot)$: because of the unbalanced distribution of the ciphertext in this cipher, it may be possible for an attacker to guess some bits of the current chaotic state with high probability. Assume $f_{be}\left(x_0^{(i)}\right)$ extracts 16 bits directly from the binary representation of $x_0^{(i)} = 0.b_1 b_2 \cdots b_j \cdots$, we can explain such insecurity about information leaking of $x_0^{(i)}$.

As we know, although the ciphertext units C_i are 16-bit integers, the probability of $C_i \geq 2^{12}$ is very small (please see Fig. 3 of [84] and Fig. 1 of [104]). Hence, if we assume that the four most significant bits are all zeros, such an assumption will be true with high probability, i.e., 4-bit information of $x_0^{(i)}$ is leaked from $f_{be}\left(x_0^{(i)}\right)$. For $i = 1$, such information can be then used to exhaustively search $F^{C_1}(x_0)$ with a complexity less than the complexity of exhaustive attack to x_0. Once $F^{C_1}(x_0)$ is obtained by the attacker, he can use it to decrypt any cipher-unit that is not smaller than C_1.

The above analysis shows that $f_{be}\left(x_0^{(i)}\right)$ should not leak information of $x_0^{(i)}$, that is to say, it should be cryptographically hard for an attacker to derive any useful information about $x_0^{(i)}$ from $f_{be}\left(x_0^{(i)}\right)$. In the following we will give two classes of such bit extracting functions, as examples to demonstrate how to make $f_{be}\left(x_0^{(i)}\right)$ cryptographically strong. With the two classes of functions, it is rather difficult for an attacker to derive any information about $x_0^{(i)}$ from partial bits of $f_{be}\left(x_0^{(i)}\right)$.

The first class is

$$f_{be}\left(x_0^{(i)}\right) = f'_{be}\left(\bigoplus_{j=0}^{C_1+\cdots+C_i} F^j(x_0)\right) = f'_{be}\left(x_0 \oplus F(x_0) \oplus \cdots \oplus x_0^{(i)}\right), \tag{5.8}$$

where $f'_{be}(x)$ can be *any* function that extracts 16 bits from the binary representation of x. Using this class of bit extracting functions, an attacker can only get some information about $\bigoplus_{j=0}^{C_1+\cdots+C_i} F^j(x_0)$. Consider $C_i \geq N_0 = 250$, it is almost impossible for an attacker to get any useful information about $x_0^{(i)}$ from $f_{be}\left(x_0^{(i)}\right)$.

The second class is

$$f_{be}\left(x_0^{(i)}\right) = \sum_{j=0}^{15} 2^j \cdot b\left(F^j(x_0^{(i)}), \left\lfloor F^{j+m}(x_0^{(i)}) \cdot 2^n \right\rfloor \bmod 16\right), \tag{5.9}$$

where $m \geq 1, n \geq 4$ and $b(x, j) = \lfloor x \cdot 2^j \rfloor \bmod 2$. In this class of bit extracting functions, all 16 bits are extracted from different chaotic states, and the positions of extracted bits are determined by chaotic states that are different from the ones the bits are extracted from ($m \geq 1$). Apparently, this class can be easily extended to many variants, for example, we can change $j + m$ to $j - m$ or change the definition of $b(\cdot)$. Also, we can combine the above two classes to realize more complex bit extracting functions, which will further enhance the security.

What's more, by cancelling non-uniformity of the ciphertext in the this modified Baptista's cipher, another two methods can also be used to avoid the information leaking of $f_{be}\left(x_0^{(i)}\right)$ effectively. With the following methods, bit extracting function can be *freely* selected.

- *Method 1*. Using the modified Baptista's cipher proposed by W.-K. Wong et al. in [104]: the distribution of the ciphertext has been enhanced to be nearly uniform, then the information leaking becomes practically impossible (see Fig. 2 of [104]).
- *Method 2*. Introducing compression mechanism: after $\widetilde{C}_i$ is obtained, compress it with any lossless entropy compression algorithm (such as Huffman compression algorithm[216]) to cancel the information redundancy (i.e., to make the distribution of $\widetilde{C}_i$ nearly uniform) and then mask the compressed $\widetilde{C}_i$ with $f_{be}\left(x_0^{(i)}\right)$. Here, please note that the smaller $\widetilde{C}_i$ is, the larger the occurrence probability will be, and the smaller the length of the compressed C_i will be, i.e., the less bits will be needed to mask the compressed $\widetilde{C}_i$. Because the size of ciphertext will be time-variant, an extra data may be needed to indicate the size of each ciphertext unit.

From the above discussions, we can see that our modified M. S. Baptista's cipher is more secure than the original M. S. Baptista's cipher. To break the modified chaotic cipher, the initial condition x_0 and the control parameter r of the chaotic systems must be broken firstly to get $x_0^{(i)}$, which just means exhaustive attack of the secret key. Of course, there still exists one defect: the encryption/decryption speed is relatively (but not much) slower than the original one.

§5.7 Conclusion

In this chapter, we show our studies on searching based chaotic ciphers. Some in-depth investigations on G. Alvarez et al.'s attacks of E. Alvarez et al.'s cipher and Jakimoski-Kocarev attack of M. S. Baptista's cipher are made. Two remedies are respectively proposed to enhance E. Alvarez et al.'s cipher against G. Alvarez et al.'s attacks, and to enhance M. S. Baptista's cipher against Jakimoski-Kocarev attack. Discussion on my remedy on M. S. Baptista' cipher introduces an interesting cipher with probabilistic-decryption feature, which has similar property with visual cryptography and may be extended to fulfill special need in conventional cryptography.

Although the security of the original two searching based ciphers can be improved, the encryption speed cannot be essentially improved much because of the time-consuming search procedure. Generally speaking, the encryption speed of all searching based ciphers is much slower than most conventional ciphers. It is a remained weakness in our modified ciphers. I will try to find solutions to this problem in future research, but perhaps any solution will lead to entirely different encryption structure so that the cipher is no longer a searching based cipher.

Chapter 6
Cryptanalysis of S. Papadimitriou et al.'s Digital Chaotic Cipher

§6.1 Introduction

As an active research group on chaotic ciphers, S. Papadimitriou et al. have made a number of works in this area[33, 36, 106, 219]. In [106], they proposed a new digital chaotic cipher, which is a probabilistic symmetric block cipher based on chaotic systems of difference equations. In this chapter, we will point out some problems with this chaotic cipher. Some problems make S. Papadimitriou et al.'s chaotic cipher unpractical and insecure, and other ones show that some remedies should be adopted to improve the performance of this chaotic cipher.

This chapter is organized as follows. In the next section, we will give a brief introduction of S. Papadimitriou et al.'s chaotic cipher. §6.3 gives detailed analyses and discussions about the above problems with S. Papadimitriou et al.'s chaotic cipher. A concrete example is given in §6.4 to show the correctness of our theoretical analyses. Several positive points about S. Papadimitriou et al.'s cipher is given in §6.5. The conclusion is given in the last section. A lengthy proof about my statement in §6.3.2 is given in Appendix of this chapter.

§6.2 S. Papadimitriou et al.'s Chaotic Cipher

For the sake of readers' convenience, in this section, we briefly introduce S. Papadimitriou et al.'s chaotic cipher and the analyses given in their paper. For more details about the original authors' descriptions and analyses, please refer to their own paper.

S. Papadimitriou et al.'s cipher is a probabilistic symmetric cipher that encrypts d-bit plaintexts into e-bit ciphertexts ($e > d$). Its encryption and decryption procedure are described below. Here, please note that we rearrange the processing steps given in [106] to obtain clearer description.

Encryption:

1. Given a (or multiple) chaotic system to generate a normalized (scaled into the unit $[0, 1]$) chaotic orbit $\{x(n)\}_{i=1}^{\infty}$.

2. Use $\{x(n)\}_{i=1}^{\infty}$ to construct a *virtual state space*, i.e., a list of 2^d *virtual attractors* containing 2^e *virtual states* $1 \sim 2^e$ as follows: search $1 \sim 2^e$ in the sequence

$\{\text{round}(x(n)\cdot 2^e)\}_{i=1}^{\infty}$ until all integers are found with shuffled orders; select 2^d states as the virtual attractors and (pseudo-randomly) allocate the left $2^e - 2^d$ states into the 2^d attractors.

3. Associate each virtual attractor V_a with a message symbol by means of a permutation matrix $\mathbf{P}$. Here, $\mathbf{P}$ is a zero-indexed 1×2^d vector* whose elements are 2^d shuffled virtual attractors between 1 and 2^e.

4. Encrypt a plain-symbol $M_c = 0 \sim 2^d - 1$ as follows: firstly map M_c to a corresponding virtual attractor by $V_a = \mathbf{P}[M_c]$, then pseudo-randomly select a virtual state S_{V_a} allocated into V_a as the ciphertext. Apparently, the last step causes this cipher to be a probabilistic symmetric block cipher.

Decryption:

1. Reconstruct the same *virtual state space* using the same method described in step 1 and step 2 of encryption.

2. Determine $\mathbf{P}$'s "inverse matrix" $\mathbf{P}^{-1}$, which is a one-indexed 1×2^d vector whose elements are $0 \sim 2^d - 1$. $\mathbf{P}^{-1}$ should satisfy the following requirement: $\forall M_c = 0 \sim 2^d - 1$, $\mathbf{P}^{-1}[\mathbf{P}[M_c]] = M_c$.

3. Retrieval the attractor V_a in which the ciphertext S_{V_a} is allocated, and then recover the plain-symbol by $M_c = \mathbf{P}^{-1}[V_a]$.

Assume the association between 2^e virtual states and 2^d virtual attractors as a surjective (multiple-to-one) map $\boldsymbol{F}_v : \boldsymbol{V}_s \to \boldsymbol{V}_a$, where V_s, V_a respectively represent the set of all virtual states and the set of all virtual attractors. Based on $\boldsymbol{F}_v$, we can conceptually denote S. Papadimitriou et al. cipher as follows: encryption – $S_{V_a} = \boldsymbol{F}_v^{-1} \circ \mathbf{P}(M_c)$, decryption – $M_c = \mathbf{P}^{-1} \circ \boldsymbol{F}_v(S_{V_a})$. Because $\boldsymbol{F}_v^{-1}$ is not unique, the encryption is probabilistic, while the decryption is deterministic since $\mathbf{P}^{-1} \circ \boldsymbol{F}_v$ is unique.

S. Papadimitriou et al. adopted the following chaotic systems with difference equations to construct the normalized chaotic orbit:

$$i = 1 \sim K: \qquad x_i(n+1) = \sum_{j=1}^{K} a_{ij} \cdot f_i\left(b_{ij} \cdot x_j(n) \bmod R_i + L_i\right), \tag{6.1}$$

where $R_i = U_i - L_i$ and $[L_i, U_i]$ is the definition domain of f_i, and the functions f_i $(i = 1 \sim K)^\dagger$ are suggested being *piecewise linear* functions with N break points, because the piecewise linearity is helpful to simplify the implementation and can

*Although S. Papadimitriou et al. call $\mathbf{P}$ a vector, we think it is more of a bijective function mapping message symbols to virtual attractors.

†In [106], the authors mistook $f_i, i = 1 \sim K$ for $f_i, i = 1 \sim K - 1$.

ensure perfect properties of the above chaotic systems. Since there are K chaotic sub-systems in total, any one sub-orbit or the combination of some of them may be available to generate virtual state spaces for encryption/decryption*.

On the security of the chaotic cipher, two possible attacks are analyzed in [106]: 1) directly reconstructing the virtual state space; 2) accurately mimicking the chaotic dynamics that leads to reconstruction of the virtual state space. The complexity of the first attack is calculated based on the estimated number of all possible virtual state spaces, which is derived to be $(k!)^m \cdot k^{n-k \cdot m}$, where $k = 2^d$ is the number of all virtual attractors and $n = 2^e$ is the number of all virtual states (m is the least number of the virtual states allocated in each virtual attractor)†. The complexity of the second attack can be calculated using the similar method given in another two S. Papadimitriou et al.'s paper [33, 219].

Other merits claimed by S. Papadimitriou et al. include: 1) piecewise linearity of the selected chaotic system makes the computational complexity rather sufficient and the cipher easy to be scaled; 2) experiments show that this cipher can run much faster than many other conventional ciphers, such as DES, IDEA and RC5.

§6.3 Problems with S. Papadimitriou et al.'s Chaotic Cipher

In this section, we will point out and give detailed discussions on the following problems with S. Papadimitriou et al.'s chaotic cipher.

1. Paradox exists between the practical implementation and high security: the size of the ciphertext and the plaintext (d and e) should be large enough to ensure high security, while it should be small enough to enable practical implementation.

2. The value of the number of all possible virtual states is deduced by a wrong way.

3. The security analysis given in [106] is inadequate and the security to exhaustive attack is overestimated.

4. The merit of fast encryption speed is dependent on the defect about the values of d and e.

*This issue is not explicitly mentioned by S. Papadimitriou et al. in their paper, but the first sub-orbit is used in their C++ codes. The codes are available upon request to S. Papadimitriou's e-mail address: stergios@heart.med.upatras.gr. Also, I have a local copy of the codes.

†In [106], N, K, M are used here, among which N, K are easily confused with the number N and K in Eq. (6.1). To avoid such a confusion, we use the lowercase formats n, k, m to replace N, K, M in [106].

5. When digital chaotic systems are realized in finite precision, the dynamical degradation will arise and some remedy should be employed to improve it.

6. No explicit instructions are given to show how to select the 2^d virtual attractors from the 2^e integers, how to allocate the 2^e virtual states into the 2^d attractors, and how to generate the permutation matrix $\mathbf{P}$.

§6.3.1 Paradox on Values of d and e

In S. Papadimitriou et al.'s cipher, the plaintext size is d and the ciphertext size is e. To provide high security, d and e should be large enough. However, we note that d and e must be small enough to make the construction and storage of the virtual state space practical, considering the following two facts: i) the time consuming on the construction of virtual states space is $O(2^e)$; ii) the number of required memory units to store the constructed virtual state space is $O(2^e)$. Apparently, e cannot be too large, generally, $e > 30$ may be unpractical for the implementation on a PC ($2^{30} = 1$G, so large a number will make the construction of the virtual state space **very very slow** and the storage **impossible** for a PC with less memory than 1G Bytes). In addition, since d and e will not be too large, an eavesdropper can exactly reconstruct the virtual state space to break the cipher once he gets $O(2^e)$ ciphertexts and the corresponding plaintexts. That is to say, the cipher is insecure to known-plaintext, chosen-plaintext and chosen-ciphertext attack[144]. In weaker conditions, it may be possible for an eavesdropper to deceive legal users with faked ciphertexts, if he can get enough (but less than 2^e) plaintexts and the corresponding ciphertexts.

Actually, in conventional cryptography, the kernel task is to design nonlinear bijective maps from the plaintexts to the ciphertexts controlled by a single secret key, where the bijective nonlinear maps play the same role as the virtual state space used in [106]. Generally speaking, the nonlinear maps used to encrypt the plaintext and decrypt the ciphertext are represented by the nonlinear operations of the secret keys, not pre-calculated in advance like the virtual state space in [106]. Then why not directly use pre-calculated and pre-stored bijective maps? It is because that the representation and storage of the map will become entirely unpractical if the size of the plaintext and the ciphertext is large enough. For example, let us consider DES: the size of the plaintext/ciphertext is 64, it is obviously impossible to represent and store a map from 2^{64} plaintexts to 2^{64} ciphertexts with limited memory units ($2^{64} = 16$GG!!). Here, we would like to cite what B. Schneier written in his well-sold book "Applied Cryptography"[144, §14.10.7]: *it will be rather easy to design a secure block cipher if you have a HUGE memory device to store HUGE-size S-Boxes*. From such a viewpoint, the basic idea of virtual state

space used in S. Papadimitriou et al.'s cipher is **unpractical** and **insecure**.

§6.3.2 Wrong Deduction of the Number of All Possible Virtual State Spaces

To estimate the security of the proposed cipher to the attack of reconstructing the virtual state space, the number of all possible spaces is deduced to be $(k!)^m \cdot k^{n-k\cdot m}$ by S. Papadimitriou et al. Based on the above result, it is claimed that the security of the proposed cipher is much higher than many other traditional ciphers, such as DES, IDEA and RSA.

In this subsection, we point out that the deduction given in [106] is not correct and the right number is not $(k!)^m \cdot k^{n-k\cdot m}$. Carefully investigate the deduction procedure given in [106], the reason can be explained by the following two problems: 1) the number may be **underestimated** since different mk states may be selected in the first stage; 2) the number may be **overestimated** since some placements are repeatedly enumerated. For the second problem, we can give one example. The following two placements A and B are same and will be repeatedly enumerated by S. Papadimitriou et al.'s deduction: all states are allocated into the same attractors for placement A and B, but a state S_{V_a} is allocated in attractor V_a in the **first** stage for placement A, and S_{V_a} is allocated in attractor V_a in the **second** stage for placement B. Since the two problems influence the result in paradoxical ways, the right number may be smaller or larger than $(k!)^m \cdot k^{n-k\cdot m}$.

In the following context, we try to solve this problem in another way. Please note such a fact: the orders of all virtual states allocated into a same attractor cannot influence the decryption of one ciphertext, although it may make the ciphertexts different for a same plaintext. Hence, the number of all possible virtual state spaces can be re-described as the solution of the following combinatorial problem: *place n **different** balls into k **different** boxes with at least m balls in each box ($n \geq mk$), how many possible placements are there?*

Then what is the right solution to the above combinatorial problem? In fact, to the best of my knowledge, no explicit solution to this problem has been reported till now, except some special ones (the Stirling's number of the second kind is the special case when $m = 1$[220]). Assume the number is $g(n)$, the best solution to this problem is a recursive one:
When $n = mk$:

$$g(n) = \binom{mk}{m, m, \cdots, m} = \frac{(mk)!}{(m!)^k}. \tag{6.2}$$

When $n > mk$:

$$g(n) = \frac{\sum_{t=mk}^{n-1} g(t)\left(k \cdot \binom{n+m-1}{t} - \binom{n+m-1}{t-1}\right)}{\frac{n-mk}{k} \cdot \binom{n+m-1}{n}}. \tag{6.3}$$

The deduction of the above solution is lengthy, so I place it in appendix of this chapter. The above solution is utterly different from the one given in [106]. For example, when $n = mk$, the right number should be $\frac{(mk)!}{(m!)^k}$, but the number is $(k!)^m$ as the deduction in [106]. In many cases, the number derived by S. Papadimitriou et al. is **smaller** than the actual one. Then can we say that the security of S. Papadimitriou et al.'s cipher may also be underestimated sometimes? The answer is **negative**, which will be explained in the next subsection with more details.

§6.3.3 Inadequate Security Analysis

In the last subsection, we have shown that the number of all possible virtual state spaces $g(n)$ should be the value expressed by Eq. (6.2) and (6.3), not $(k!)^m \cdot k^{n-k\cdot m}$ given in [106]. In this subsection, we will point out that the value of $g(n)$ and the number of all possible secret keys cannot be directly used to show the high security of the proposed cipher as S. Papadimitriou et al. did in [106]. It is a natural result of the following four facts **F1** to **F4**.

F1) *Most virtual state spaces are too "similar" to ensure the high security of the chaotic cipher*. To quantitatively measure the similarity of two different virtual state spaces A, B, we firstly give a notation $d(A, B)$ called the **distance** of A and B as follows: $d(A, B) = \sum_{i=1}^{n} \mathrm{Com}(A_i, B_i)$, where A_i, B_i are the virtual attractors containing the i^{th} virtual states in A, B, and

$$\mathrm{Com}(A_i, B_i) = \begin{cases} 1, & A_i \neq B_i \\ 0, & A_i = B_i \end{cases}. \tag{6.4}$$

Here, $d(A, B) = 1 \sim n$ represents the number of virtual states allocated into different attractors in A, B. Apparently, the smaller $d(A, B)$ is, the more similar the two virtual state spaces A and B will be.

As a result of the property of the distance $d(A, B)$, similar virtual state spaces will generate similar ciphertexts with uniformly distributed plaintexts. Thus, an eavesdropper can use a similar virtual state space instead of the real one to decrypt most plaintexts (the more similar the used one is to the real one, the more plaintexts will be decrypted). To obtain high enough security, the distance between any two available virtual state spaces A,B should be large enough

($d(A,B) = n$ will be really perfect and $d(A,B) \geq n/2$ may be acceptable in many cases), but the number of such "**good**" virtual state spaces will be **much much smaller** than the number given by Eq. (6.2) and (6.3).

F2) *Not all possible virtual state spaces can be constructed with the chaotic system (6.1).* Once the chaotic orbit $\{x(i)\}_{i=1}^{\infty}$ and the algorithm to construct the virtual state space is given, the generated virtual state space will be uniquely determined. The above fact means that the number of all possibly generated virtual state spaces is also controlled by the number of all possible chaotic orbits as well, not only by Eq. (6.2) and (6.3). Then what is the number of all possible chaotic orbits? Apparently, it is determined by the number of all possible secret keys, i.e., all possible control parameters and initial conditions.

In S. Papadimitriou et al.'s cipher, the following control parameters of Eq. (6.1) are used as the secret keys*: $a_{ij}, b_{ij} (i,j = 1 \sim K)$, $R_i, L_i (i = 1 \sim K)$ and NK break point values of $f_1 \sim f_K$ (only N break point values if $f_1 = f_2 = \cdots = f_K$)†. Assume the computed sensitivities of the above parameters are all 2^{-L} (L is the adopted finite computing precision) and all parameters are confined in $[0,1]$, we can roughly calculate the number of all possible secret keys‡: $\mathcal{N}_{\mathcal{K}} = (K^2 + 2K) \cdot 2^L + K \cdot \prod_{i=0}^{N-1}(2^L - i)/N!$. Generally, $2 < N \ll 2^L$ and $2 < K \ll 2^L$, then $\mathcal{N}_{\mathcal{K}} \approx K \cdot 2^{LN}/N!$ (when $f_1 = f_2 = \cdots = f_K$, $\mathcal{N}_{\mathcal{K}} \approx 2^{LN}/N!$).

F3) *Different secret keys may generate the same virtual state space.* This fact is obviously right if $\mathcal{N}_{\mathcal{K}} > g(n)$. Together with the above fact **F3**, we can see the upper bound of the security of the proposed chaotic cipher should be $\min(g(n), \mathcal{N}_{\mathcal{K}})$. Thus, although $g(n)$ may be rather HUGE when $n = 2^e$ and $m = 2^d$ are large enough ($d = e = 8$ may be OK), the actual security of S. Papadimitriou et al.'s cipher will be limited by $\mathcal{N}_{\mathcal{K}}$. From the approximate value of $\mathcal{N}_{\mathcal{K}}$ derived in the last paragraph, the key entropy of S. Papadimitriou et al.'s cipher to exhaustive attack will be about $LN - \log_2(K/N!)$ in general cases, or even smaller than $LN - \log_2(K/N!)$ if d and e are small enough to make $g(n) < \mathcal{N}_{\mathcal{K}}$.

F4) *S. Papadimitriou et al.'s chaotic cipher is insecure to the known-plaintext, chosen-plaintext and chosen-ciphertext attacks, because of the defect about the small values of d and e.* This issue has been discussed in §6.3.1. We can see the key entropy of S. Papadimitriou et al.'s cipher to the three attacks will be e, generally, which is much smaller than $LN - \log_2(K/N!)$.

*The initial conditions are not involved as part of the secret keys in [106]. 0.1 is used to initialize $x_1(0) \sim x_K(0)$ in S. Papadimitriou et al.'s C++ codes.

†In [106], the number of break point values of f_i are denoted by N in Sec. 2.2 and by n in Sec. 3. In this dissertation, we use N at all time.

‡S. Papadimitriou et al. didn't give the deduction of $\mathcal{N}_{\mathcal{K}}$ in [106] and only referred the readers to their another two previous papers [33, 219]. Here, we use a somewhat different way to calculate the value of $\mathcal{N}_{\mathcal{K}}$.

§6.3.4 Other Problems

The dependence of the perfectly fast encryption speed on the essential defect about the values of d and e. In Table 2 of [106], a comparison of the encryption speed of the proposed chaotic cipher with some traditional ciphers is given on a Celeron 433 MHz PC with 96 MB RAM. S. Papadimitriou et al.'s chaotic cipher can run at a very high speed 327.2 Mbps, which is much faster than other ones. The perfectly fast encryption speed can be explained by the following fact: once the virtual state space has been constructed, the encryption and decryption procedure (the last step) can be realized by simple *Look-Up-Table* operations. But please keep in mind that this merit owes to the defect that the whole virtual state space must be firstly constructed and then stored in memory, which makes the cipher unpractical and insecure as we have mentioned in §6.3.1.

The lack of explicit instructions on how to select the 2^d *virtual attractors, how to allocate the* 2^e *virtual states into the* 2^d *attractors and how to generate* $\mathbf{P}$. This problem is not so serious since many different pseudo-random coding algorithms can be used to do the above three operations. Of course, different algorithms may lead to different performances on the pseudo-randomness of the selection of the 2^d virtual attractors, the allocation of the 2^e virtual states the permutation matrix $\mathbf{P}$. In addition, if we know the algorithm used in the cipher, it may be possible to analyze the generated virtual state space. Such analysis may be useful to develop some new attack whose complexity is less than the exhaustive attack's (at least under some special conditions). Some further research should be made to investigate this issue.

Dynamical degradation of chaotic systems realized in finite computing precision. As we discussed in §2.5.1, such dynamical degradation exists for any digital chaotic system and must be remedied with some countermeasures.

§6.4 A Concrete Example

To emphasize the paradox between security and feasibility of S. Papadimitriou et al.'s chaotic cipher, now let us give a concrete example for further explanation. Considering the chaotic system is just used to generate the virtual states with higher key entropy, we will use Logistic map $F(x) = 4x(1-x)$ instead of the chaotic system suggested in [106], which will not make essential influence on the performance of this cipher. Assume $d = 6, e = 8$, the secret key is the initial condition of Logistic map $x_0 = 0.1111$. Without loss of generality, assume $m = 3$, and the 2^d virtual attractors, the allocations of other $2^e - 2^d$ virtual states (i.e.,

the map $\boldsymbol{F}_v$) and the permutation matrix $\mathbf{P}$ are all pseudo-randomly generated* with the control of the embedded system function `rand` initialized by a (secret or public) seed $s = 0.2222$. Here, please note neither x_0 or s is specially chosen to support our negative result. The constructed map $\boldsymbol{F}_v$ (i.e., the association between the virtual states and the virtual attractors) and the permutation matrix $\mathbf{P}$ are respectively shown in Figure 6.1 and 6.2.

For such a encryption system, if we can get enough known/chosen plaintext/ciphertext pairs, it is possible to obtain the unique decryption function $\mathbf{P}^{-1} \circ \boldsymbol{F}_v$. Since d, e is not too large, we can store this function as a look-up table in the computer to decrypt all future ciphertexts. What about the number of required known/chosen plaintexts? In Figure 6.3, under the assumption that the plaintext is uniformly distributed in the discrete set $\{0, 1, \cdots, 2^d - 1\}$, we give the experimental result of the relation between the number of obtained virtual states/attractors and the number of known/chosen plaintexts. We can see $O(2^e)$ plaintexts are enough to obtain all 2^e virtual states (i.e., all possible ciphertexts) and $O(2^d)$ plaintexts are enough to obtain all 2^d virtual attractors. What $O(2^e)$ plaintexts mean? Consider the plaintexts are 6-bit numbers, $O(2^8)$ plaintexts mean only about 192 bytes, which approximates to the length of a long English sentence. Once all 2^e virtual states are obtained, we can reconstruct the ciphertext-plaintext map (i.e., the decryption function) $\mathbf{P}^{-1} \circ \boldsymbol{F}_v$. Apparently, such a security defect is induced by the small values of d, e. But if we increase d, e to resist such attacks, the construction and storage of $\boldsymbol{F}_v$ will become impractical.

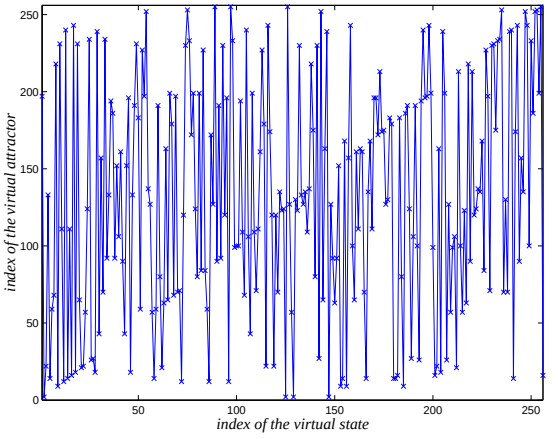


Figure 6.1: The association map $\boldsymbol{F}_v$

Finally, let us see the number of all possible maps $\boldsymbol{F}_v$. When $n = 2^e = 256, m = 3, k = 2^d = 64$, the number of all possible placements of n balls

*As we have pointed out in §6.3.4, no explicit instructions are given to direct how to generate them.

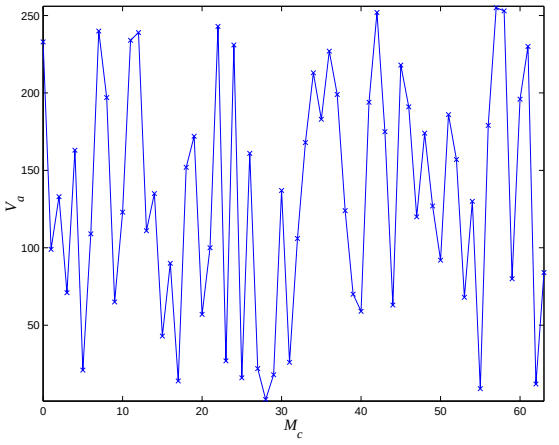


Figure 6.2: The permutation matrix **P**

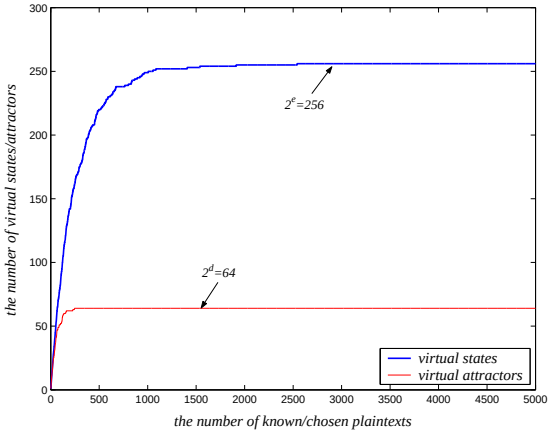


Figure 6.3: The number of obtained virtual states from known/chosen plaintexts

in k boxes (each one at least m balls) is so great that it even cannot be calculated with most scientific computing software: $g(n) \gg 10^{308} \approx 2^{1023}$. However, the number of all possible initial conditions x_0 is generally much much smaller than $g(n)$. When x_0 is a IEEE-standard double precision (64-bit) floating-point decimal[217], then $\mathcal{N}_k = 2^{62} \ll g(n)$. Thus, the complexity against brute force attack will be $O(\min(g(n), \mathcal{N}_k)) = O(2^{62})$. However, from the analysis in §6.3.1 and the above experimental data in this subsection, the complexity against known/chosen plaintext attack is only $O(2^e) = O(2^8) \ll O(2^{62})$.

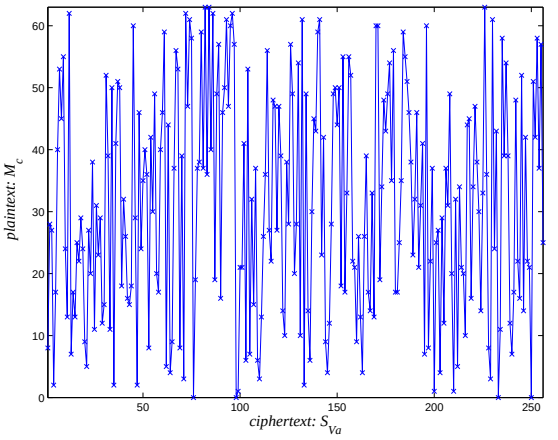


Figure 6.4: The reconstructed ciphertext-plaintext map $\mathbf{P}^{-1} \circ \boldsymbol{F}_v$ with $O(2^e)$ known/chosen plaintexts

§6.5 Positive Points about S. Papadimitriou et al.'s Chaotic Cipher

Although S. Papadimitriou et al.'s chaotic cipher has some problems and its general structure is not suitable as a basis to construct more secure chaotic block ciphers, some fundamental ideas used in the cipher may still be helpful in chaotic cryptography.

One useful point is about the possibility to change S. Papadimitriou et al.'s chaotic cipher from a block cipher to a stream cipher, which may disable the attacks based on the re-construction of the virtual attractors list and the permutation matrix $\mathbf{P}$ (via known/chosen plaintexts). A possible method is to generate time-variant permutation matrix $\mathbf{P}$, or use a stream sub-cipher to confuse the ciphertext of the S. Papadimitriou et al.'s chaotic cipher. Applications of such an idea in the design of digital chaotic ciphers have been discussed in Chap. 2.

Another point is the idea to construct virtual state space from a chaotic orbit, which can be extended as a new way to generate nonlinear $n \times m$ S-boxes without trapdoors[144, 145]. Apparently, such chaotic S-Boxes can be dependent on the secret key, and then be incorporated into some conventional key-driven ciphers to construct new chaos based ciphers. In fact, such cryptosystems based on chaotic S-Boxes have been proposed by some researchers[55, 105, 108, 112], but more detailed studies should be done to analyze the performance of such ciphers.

§6.6 Conclusion

In this chapter, we point out some defects of S. Papadimitriou et al.'s chaotic cipher proposed in [106]: 1) d and e are too small to ensure both practical implementation and high security; 2) the deduction of the number of all possible virtual state spaces is wrong; 3) inadequate analysis leads to overestimated security; 4) fast encryption speed is the result of the first defect; 5) dynamical degradation of digital chaotic systems should be remedied; 6) no detailed instructions about the construction of the virtual state space are given.

Generally speaking, because of the small values of d and e, S. Papadimitriou et al.'s chaotic cipher is unpractical and insecure to known/chosen-plaintext and chosen-ciphertext attack. Its merit of fast encryption speed may disappear if the defect about d and e is cancelled. In addition, from our discussions in §6.3.3, the security of the cipher is not so high as analyzed in [106], and the key entropy to exhaustive attack will be not larger than $LN - \log_2(K/N!)$.

Appendix: The Recursive Solution of the Combinatorial Problem in §6.3.2

Here, we give the deduction of Eq. (6.2) and (6.3).

Assume $g(n)$ is the number of all possible placements determined by n. Because

$$(x_1 + x_2 + \cdots + x_k)^n = \sum_{a_1+a_2+\cdots+a_k=n} \binom{n}{a_1, a_2, \cdots, a_k} x_1^{a_1} x_2^{a_2} \cdots x_k^{a_k}, \tag{6.5}$$

We have

$$g(n) = \sum_{\substack{a_1+a_2+\cdots+a_k=n \\ a_i \geq m}} \binom{n}{a_1, a_2, \cdots, a_k}. \tag{6.6}$$

Consider the following exponential generating function:

$$\begin{aligned} \sum_{n \geq mk} g(n) \frac{x^n}{n!} &= \sum_{n \geq mk} \left(\sum_{\substack{a_1+a_2+\cdots+a_k=n \\ a_i \geq m}} \binom{n}{a_1, a_2, \cdots, a_k} \right) \frac{x^n}{n!} \\ &= \sum_{n \geq mk} \left(\sum_{\substack{a_1+a_2+\cdots+a_k=n \\ a_i \geq m}} \frac{n!}{a_1! \cdot a_2! \cdots a_k!} \right) \frac{x^{a_1+a_2+\cdots+a_k}}{n!} \end{aligned}$$

$$= \left(\sum_{a\geq m}\frac{x^a}{a!}\right)^k. \tag{6.7}$$

Consequently, $\left(\sum_{a\geq m}\frac{x^a}{a!}\right)^k$ is the generating function of $g(n)$.

Apparently, it is hard to derive the explicit equation of $g(n)$ denoted by n, m, k, so let us investigate the recursive expression of $g(n)$.

Rewrite Eq. (6.7) as $\sum_{i\geq mk} g(i)\frac{x^i}{i!} = \left(\sum_{a\geq m}\frac{x^a}{a!}\right)^k$, and solve the derivatives of both sides, we can have:

$$\sum_{i\geq mk-1} g(i+1)\frac{x^i}{i!} = k\cdot\left(\sum_{j\geq m}\frac{x^j}{j!}\right)^{k-1}\cdot\left(\sum_{j\geq m-1}\frac{x^j}{j!}\right), \tag{6.8}$$

Multiply the both sides of the above equation by $\sum_{j\geq m}\frac{x^j}{j!}$,

$$\begin{aligned}\left(\sum_{j\geq m}\frac{x^j}{j!}\right)\cdot\left(\sum_{i\geq mk-1} g(i+1)\frac{x^i}{i!}\right) &= k\cdot\left(\sum_{a\geq m}\frac{x^a}{a!}\right)^k\cdot\left(\sum_{j\geq m-1}\frac{x^j}{j!}\right)\\ &= k\cdot\left(\sum_{a\geq mk} g(a)\frac{x^a}{a!}\right)\cdot\left(\sum_{j\geq m-1}\frac{x^j}{j!}\right)\end{aligned}. \tag{6.9}$$

The left hand side (LHS) of Eq. (6.9) is

$$\begin{aligned}\text{LHS} &= \sum_{i\geq mk+m-1}\left(\sum_{\substack{s+t=i\\ s\geq m\\ t\geq mk-1}}\left(\frac{g(t+1)}{s!t!}x^i\right)\right)\\ &= \sum_{i\geq mk+m-1}\left(\sum_{\substack{s+t=i\\ s\geq m\\ t\geq mk-1}} g(t+1)\binom{i}{s}\right)\frac{x^i}{i!}.\end{aligned} \tag{6.10}$$

The right hand side (RHS) of Eq. (6.9) is

$$\begin{aligned}\text{RHS} &= k\cdot\sum_{i\geq mk+m-1}\left(\sum_{\substack{s+t=i\\ s\geq m-1\\ t\geq mk}}\left(\frac{g(t)}{s!t!}x^i\right)\right)\\ &= \sum_{i\geq mk+m-1} k\cdot\left(\sum_{\substack{s+t=i\\ s\geq m-1\\ t\geq mk}} g(t)\binom{i}{s}\right)\frac{x^i}{i!}.\end{aligned} \tag{6.11}$$

Thus, we can know the following fact: when $i \geq mk + m - 1$,

$$\sum_{\substack{s+t=i \\ s\geq m \\ t\geq mk-1}} g(t+1)\binom{i}{s} = k \cdot \sum_{\substack{s+t=i \\ s\geq m-1 \\ t\geq mk}} g(t)\binom{i}{s}. \tag{6.12}$$

Since $s + t = i$, $\binom{i}{s} = \binom{i}{t}$, then the above equation can be transformed to:

$$\sum_{t=mk-1}^{i-m} g(t+1)\binom{i}{t} = k \cdot \sum_{t=mk}^{i-m+1} g(t)\binom{i}{t}. \tag{6.13}$$

Substitute $t' = t + 1$ into the left hand side of the above equation, we can get:

$$\sum_{t'=mk}^{i-m+1} g(t')\binom{i}{t'-1} = k \cdot \sum_{t=mk}^{i-m+1} g(t)\binom{i}{t}. \tag{6.14}$$

Based on Eq. (6.14), we can derive the recursive solution of $g(n)$.
When $n = mk$:

$$g(n) = \binom{mk}{m, m, \cdots, m} = \frac{(mk)!}{(m!)^k}. \tag{6.15}$$

When $n > mk$: assume $i - m + 1 = n$, $i = n + m - 1$. Substitute $i = n + m - 1$ into Eq. (6.14), we can get:

$$\sum_{t=mk}^{n} g(t)\binom{n+m-1}{t-1} = k \cdot \sum_{t=mk}^{n} g(t)\binom{n+m-1}{t}. \tag{6.16}$$

Simplify the above equation:

$$g(n) = \frac{\sum_{t=mk}^{n-1} g(t)\left(k \cdot \binom{n+m-1}{t} - \binom{n+m-1}{t-1}\right)}{\frac{n-mk}{k} \cdot \binom{n+m-1}{n}}. \tag{6.17}$$

From Eq. (6.15) and (6.17), this problem is solved.

Chapter 7
Cryptanalysis of Two Yen-Guo's Chaotic Image Encryption Methods

§7.1 Introduction

The dramatically progress of Internet makes security of digital images more and more important since the exchanges of digital images over network occur more and more frequently. Furthermore, special and reliable security in storage and transmission of digital images is needed in many applications, such as pay-TV, medical imaging systems, military image database/communications and online image database services (for example, online personal albums), etc. In order to fulfill such a task, many image encryption methods have been proposed[18, 85, 89, 132–138, 221–225] to protect the content of digital images, but some of them[223–225] have been known to be insecure[221, 226].

As we surveyed in §2.4.6, a large number of chaotic image encryption methods[18, 85, 89, 132–138] have been proposed and many ones[132–135, 137, 138] are contributed by J.-C. Yen and J.-I. Guo (et al.). Yen-Guo's chaotic image encryption methods yield the following basic idea: a chaotic map (Logistic map is used for all Yen-Guo's chaotic image encryption methods) serves as a chaotic PRNG, and the PRNG is used to control secure permutations or substitutions of pixels. From the cryptographical point of view, most Yen-Guo cryptosystems are not secure since known/chosen plaintext attack can break them with less complexity than brute force attack (some ones can be broken with only several plain-images). In this chapter, I will introduce our cryptanalyses on two Yen-Guo chaotic image encryption methods, which are respectively called CKBA[134] (Chaotic Key-Based Algorithm) and BRIE[132] (Bit Recirculation Image Encryption). Both methods are not secure because they are not carefully designed against known/chosen plaintext attacks. The insecurity of Yen-Guo's chaotic image encryption methods implies that the gap between signal/image processing and cryptography should be bridged.

This chapter is organized as follows. In §7.2, it is shown that how CKBA and BRIE work. Cryptanalysis of CKBA is given in §7.3, where some examples are also given to show the feasibility of the proposed attacks. Some specific security defects of BRIE are discussed in §7.4. Cryptanalysis of BRIE is given in §7.5. Some experimental results are also included in §7.4 and §7.5. In §7.6 we discuss some remedies of CKBA and BRIE. The last section is the conclusion.

§7.2 Two Yen-Guo's Image Encryption Methods: CKBA and BRIE

§7.2.1 CKBA: Chaotic Key-Based Algorithm

The encryption procedure of CKBA can be briefly depicted as follows. Assume the size of the plain-image is $M \times N$. Select two bytes $key1$ and $key2$ (8 bits) and the initial condition $x(0)$ of a one-dimensional chaotic system (Logistic map) as the secret keys of the encryption system. Run the chaotic system to make a chaotic sequence $\{x(i)\}_{i=0}^{MN/8-1}$ (assume $MN|8$). Generate a pseudo-random binary sequence (PRBS) $\{b(i)\}_{i=0}^{2MN-1}$ from the 16-bit binary representation of $x(i) = 0.b(16i+0)b(16i+1)\cdots b(16i+15)$. Once $\{b(i)\}$ is generated, the encryption can start. For the plain-pixel $f(x,y)(0 \le x \le M-1, 0 \le y \le N-1)$, the corresponding cipher-pixel $f'(x,y)$ is determined by the following rule:

$$f'(x,y) = \begin{cases} f(x,y) \text{ XOR } \quad key1, b'(x,y) = 3 \\ f(x,y) \text{ XNOR } key1, b'(x,y) = 2 \\ f(x,y) \text{ XOR } \quad key2, b'(x,y) = 1 \\ f(x,y) \text{ XNOR } key2, b'(x,y) = 0 \end{cases}, \tag{7.1}$$

where $b'(x,y) = 2 \times b(l) + b(l+1)$ and $l = x \times N + y$. The decryption procedure is just like the encryption since XOR and XNOR are both involutive operations. Because not all secret keys can make well disorderly cipher-images, the basic criterion to select $key1$ and $key2$ should be satisfied: $\sum_{i=0}^{7}(a_i \oplus d_i) = 4$, where $key1 = \sum_{i=0}^{7} a_i \times 2^i$ and $key2 = \sum_{i=0}^{7} d_i \times 2^i$.

§7.2.2 BRIE: Bit Recirculation Image Encryption

The basic idea of BRIE is bit recirculation of pixels, which is controlled by a chaotic pseudo-random binary sequence. The secret keys of BRIE are two integers α, β and the initial condition $x(0)$ of a one-dimensional chaotic system (Logistic map). Assume the size of the plain-image is $M \times N$. Run the chaotic system to make a chaotic orbit $\{x(i)\}_{i=0}^{\lceil (MN+1)/8 \rceil - 1}$. Then generate a pseudo-random binary sequence (PRBS) $\{b(i)\}_{i=0}^{MN}$ from the 8-bit binary representation of $x(i) = 0.b(8i+0)b(8i+1)\cdots b(8i+7)$. For the plain-pixel $f(x,y)(0 \le x \le M-1, 0 \le y \le N-1)$, the cipher-pixel $f'(x,y)$ is determined by the following equation:

$$f'(x,y) = ROLR_p^q(f(x,y)), \tag{7.2}$$

where $p = b(N \times x + y), q = \alpha + \beta \times b(N \times x + y + 1)$ and $ROLR_p^q$ is a cyclical shift by q bits in a direction controlled by p:

$$ROLR_p^q(x = b_7 b_6 \cdots b_0) = \begin{cases} \sum_{i=0}^{7} b_i \cdot 2^{(i-q+8) \bmod 8}, & p = 0 \\ \sum_{i=0}^{7} b_i \cdot 2^{(i+q) \bmod 8}, & p = 1 \end{cases}. \tag{7.3}$$

The decryption procedure can be denoted by

$$f(x,y) = ROLR_{1-p}^q(f'(x,y)) = ROLR_p^{8-q}(f'(x,y)). \tag{7.4}$$

Apparently, BRIE is a pixel transformation cipher, i.e., the cipher-pixel at (x,y) is uniquely determined by the plain-pixel at the same position. J.-C. Yen and J.-I. Guo claimed that BRIE needs very low computation complexity, and has high security since $\{b(i)\}$ contains $MN + 1$ secure bits generated by the chaotic iterations. However, we will point out that some serious defects exist in BRIE, and that a known/chosen-plaintext attack can break it. The BRIE encrypts the plain-image column by column, which is somewhat inconvenient in practice. In this chapter, we modify BRIE to work in line mode, which will not make essential influence on its performance.

§7.3 Cryptanalysis of CKBA

§7.3.1 Ciphertext-Only Attack

J.-C. Yen and J.-I. Guo claimed that the attack complexity of CKBA is 2^{2MN} since $\{b(i)\}_{i=0}^{2MN-1}$ has $2MN$ bits. Actually, such a statement is not true because of the following fact: total $2MN$ bits are uniquely determined by the equation of the chaotic system and its initial condition $x(0)$, which has only 16 secret bits. Actually, the secret keys of CKBA are $key1$, $key2$ and $x(0)$, we can find the right secret keys with brute-force ciphertext-only attack. Since the keys totally contain $2 \times 8 + 16 = 32$ bits, the key entropy should be about 32. But not all keys can be used in CKBA because of the basic criterion $\sum_{i=0}^{7}(a_i \oplus d_i) = 4$, only $2^{16} \times 2^8 \times C_8^4 = 2^{24} \times 70 \approx 2^{30}$ keys are available in total $2^{16} \times 2^8 \times 2^8 = 2^{16}$ ones. Thus the key entropy is about $14 + 16 = 30$.

The exact attack complexity can be estimated as follows. Averagely, $MN/8$ chaotic iterations are needed for the generation of $\{b(i)\}$, and $(2^8 \times 70) \times MN = 17920 \times MN \approx 2^{14} \times MN$ XOR/XNOR operations are needed to decrypt the whole cipher-image. Assuming one chaotic iteration and one XOR/XNOR operation consumes the same time, the total attack complexity in average is about

$2^{15} \times \left(MN/8 + 2^{14} \times MN\right) \approx 2^{29}MN$, which is much smaller than 2^{2MN} when M, N are not too small ($M > 4, N > 4$). That is to say, the security of CKBA is overestimated by the authors, even under brute-force attack. Because of the rapid progress of digital computer and distributed arithmetic, the complexity with the order of $O(2^{128})$ is required for a cryptographically strong cipher, but CKBA can not provide enough security. Without loss of generality, assume $M = N = 1024 = 2^{10}$, which is the typical size of a "large" digital image, the attack complexity will be $2^{29}MN = 2^{49}$.

§7.3.2 Known/Chosen Plaintext Attack

In known-plaintext or chosen-plaintext attack, CKBA can be broken with only one plain-image and its cipher-image. Assume one knows a plain-image f and the corresponding cipher-image f' (both $M \times N$). For the plain-pixel $f(x, y)$, the cipher-pixel $f'(x, y)$ must be one of the four values: $f(x, y)$ XOR $key1$, $f(x, y)$ XNOR $key1$, $f(x, y)$ XOR $key2$, $f(x, y)$ XNOR $key2$. Since a XNOR $b = a$ XOR $\bar{b}$, $f(x, y)$ XOR $f'(x, y)$ must be one of the four values: $key1$, $\overline{key1}$, $key2$, $\overline{key2}$. Therefore, if we XOR f and f', we can get a mask image f_m, which can be used to decrypt other cipher-images encrypted with the same key K if their sizes are not larger than $M \times N$. For a plain-image whose size is larger than MN, the left MN pixels can be also decrypted directly. The computation complexity obtaining f_m is only $O(MN)$, and is independent of $key1$, $key2$ and $x(0)$.

If we want to entirely decrypt a cipher-images with larger size, the right secret key $K = \{key1, key2, x(0)\}$ must be known. Based on f_m, it is rather easy to deduce K. Because f_m only contains four possible gray values: $\{key1, \overline{key1}, key2, \overline{key2}\} = \{k_1, k_2, k_3, k_4\}$, we can find the right $key1$ and $key2$ by brute-force search. The search procedure can be described in the following steps.

- **Step 1**: Assume $key1 = k_m$ (for $m = 1 \sim 4$), and $key2 = k'_{m'}$ (for $m' = 1 \sim 2$), where k'_1 and k'_2 are the two possible values of $key2$ when $key1$ is determined (the other two are $key1$ and $\overline{key1}$).

- **Step 2**: Calculate $b'(x, y)$ for all pixels using the following rule:

$$b'(x, y) = \begin{cases} 3, f_m(x, y) = key1 \\ 2, f_m(x, y) = \overline{key1} \\ 1, f_m(x, y) = key2 \\ 0, f_m(x, y) = \overline{key2} \end{cases} . \tag{7.5}$$

- **Step 3**: Generate the chaotic orbits $\{x(i)\}_{i=0}^{MN/8-1}$ from $b'(x, y)$.

- **Step 4**: Verify whether or not $\{x(i)\}_{i=0}^{MN/8-1}$ satisfies the chaotic equation. If the answer is yes, the search procedure stops and output the current $key1$, $key2$ and $x(0)$, which are the right secret keys K. Here please note that we need not calculate the whole chaotic orbit $\{x(i)\}_{i=0}^{MN/8-1}$, just two chaotic values $x(0)$ and $x(1)$ are enough to make correct judgement.

Apparently, the computation complexity from f_m to K is chiefly determined by step 2 and 3. Generally speaking, the complexity is $O(MN)$, which approximately equals to the one obtaining f_m.

There is another possible method to decrypt any cipher-image whose size is larger than the size of the known/chosen plain-image. From the discussion in §2.5, we have known that, when chaotic systems are realized under finite computing precision L, the cycle length of the chaotic orbits will be much smaller than 2^L. For CKBA, the finite precision $L = 16$, the cycle length of each chaotic orbit will be much smaller than 2^{16}, which is not large enough in comparison with the size of many plain-images. For a 256×256 image, the total length of the chaotic orbit $\{x(i)\}$ is $MN/8 = 2^{13}$, while for many initial condition $x(0)$, the cycle length of $\{x(i)\}$ is even much smaller than 2^{13}. Consequently, it is possible to derive an entire cycle of the chaotic orbit from the known mask image f_m whose size is about $256 \times 256 = 2^{16}$, then to derive any mask image with larger size. That is to say, without extracting the right secret key K, a 256×256 mask image f_m is enough to decrypt all cipher-images. Such a result is supported by our experiments (see the next subsection and Figure 7.4). Assume the size of the larger cipher-image is $M' \times N'$, the complexity from f_m to f'_m will be $O(M'N' + MN)$, which is a little larger than the one obtaining f_m.

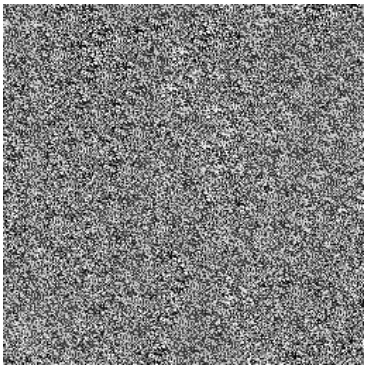
As we know, the known-plaintext and chosen-plaintext attacks will be very meaningful if a same key is used to encrypt more than one plaintexts, especially in the case that a larger number of plaintexts are all encrypted with a same key[144, 145]. For a "good" cipher, the capability to resist known-plaintext attack is very important and generally needed. It is because of the following fact: the key management will be very complex, inconvenient and inefficient in many applications, if any key must not be used to encrypt more than one plaintexts. Apparently, it is not advisable to apply CKBA to encrypt MPEG video as claimed in [134]. Once one plain-frame in the encrypted MPEG video stream is known for an illegal user, he can easily get all other plain-frames, i.e., the whole video stream.

§7.3.3 Experiments

To verify the feasibility of the above known/chosen plaintext attack, we give some experimental results in this section. Logistic map with control parameter

$r = 4$ is selected as the chaotic system and it is realized in 16-bit finite precision.

a) Lenna.bmp (256×256)



b) Encrypted Lenna.bmp

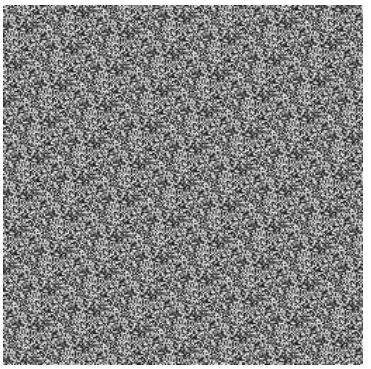
Figure 7.1: One known/chosen plain-image and its cipher-image: CKBA

For a pseudo-randomly selected key $K = \{key1, key2, x(0)\}$, one 256×256 plain-image f (Lenna.bmp) and its cipher-image f' are given in Figure 7.1. We can easily get the mask image $f_m = f$ XOR f' (Figure 7.2a).

When the key K is used to encrypt another cipher-image with identical size (see Figure 7.2b–c), the plain-image can be directly decrypted by f_m (see Figure 7.2d).

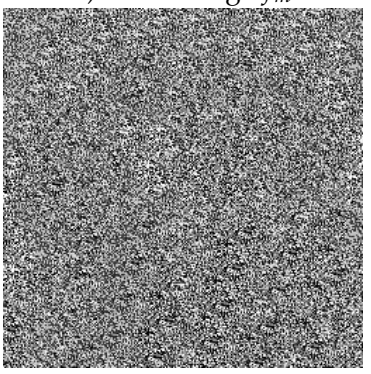
When the key K is used to encrypt a larger cipher-image (384×384, see Figure 7.3a–b), f_m can only decrypt MN pixels from the left side (see Figure 7.3c). To decrypt the whole plain-image, we can derive the right key K from f_m. Using the method described in the last subsection, we can get $key1 = 92, key2 = 36, x(0) = 12830/2^{16}$, and then the whole cipher-image can be decrypted (see Figure 7.3d).

In the last subsection, we have mentioned another method to decrypt larger plain-images. Observe f_m (Figure 7.2c) obtained from the known/chosen plain-image Lenna.bmp (256×256), we can see some obvious pattern occurs repeatedly for 9 times. It means that the cycle length of $\{x(i)\}_{i=0}^{MN/8-1}$is about $2^{16}/(8 \times 9) = 2^{16}/72$. As a result, we can easily generate the mask image f'_m for 384×384 plain-images from f_m, which is shown in Figure 7.4a. The decrypted plain-image using f'_m is shown in Figure 7.4b.



a) Mask image f_m

b) Miss.bmp (256×256)



c) Encrypted Miss.bmp

d) Decrypted Miss.bmp by f_m

Figure 7.2: Cryptanalyze Miss.bmp using f_m: CKBA

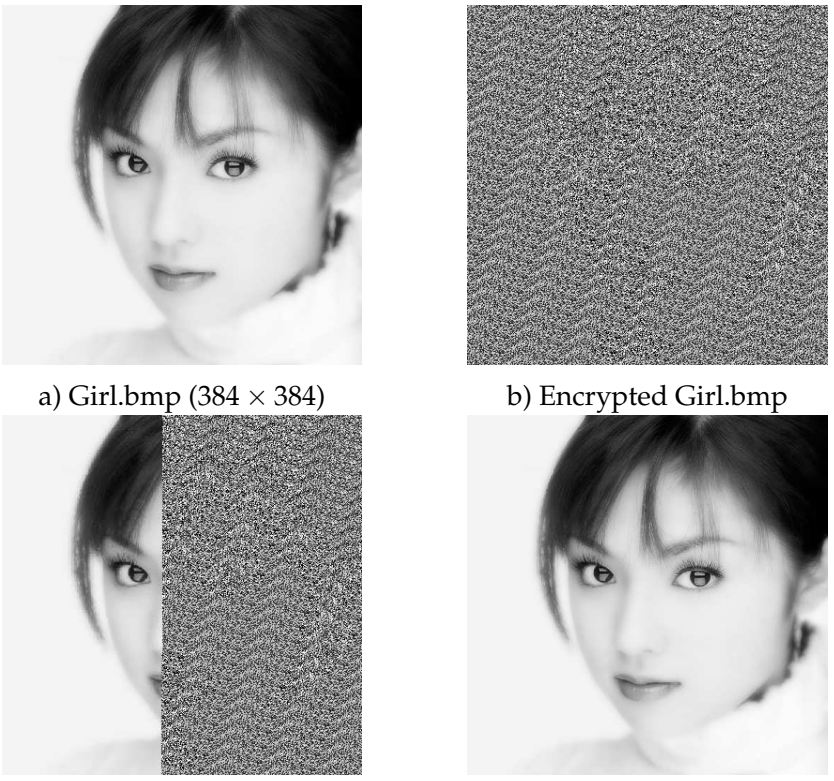
§7.4 Some Security Defects of BRIE

§7.4.1 Essential Defects of *ROLR* Operations

The *ROLR* operations controlled by pseudo-random chaotic sequence $\{b(i)\}$ are the kernel of BRIE. But *ROLR* have two essential defects when it is used in BRIE, which both lower the security of BRIE and limit its applications in practice.

1) Some plain-pixels may keep unchanged ($f'(x,y) = f(x,y)$) after encryption. If there are too many such pixels, the plain-image will roughly emerge from the cipher-image. The plain-pixels can be divided into the following four classes*. **C1**) 0, 255: $f'(x,y) \equiv f(x,y)$, $\forall \alpha, \beta$. **C2**) 85, 170: If $\alpha \bmod 2 = 0$, $f'(x,y) = f(x,y)$ when $q = \alpha$; if $\alpha + \beta \bmod 2 = 0$, $f'(x,y) = f(x,y)$ when $q = \alpha + \beta$; and if $\alpha \bmod 2 = (\alpha + \beta) \bmod 2 = 0$, $f'(x,y) \equiv f(x,y)$. **C3**) 17, 34, 51, 68, 102, 119, 136, 153, 187, 204, 221, 238: If $\alpha \bmod 4 = 0$, $f'(x,y) = f(x,y)$

*Different repeated patterns exist in the binary representation of different pixels: C1) eight repeated bits – 0 (00000000), 1 (11111111); C2) four repeated 2-bit segments – 85 (01010101), 170 (10101010); C3) two repeated 4-bit segments – 17 (00010001), etc.; C4 – no repeated pattern.



a) Girl.bmp (384 × 384) b) Encrypted Girl.bmp

c) Decrypted Girl.bmp by f_m d) Decrypted Girl.bmp by K obtained from f_m

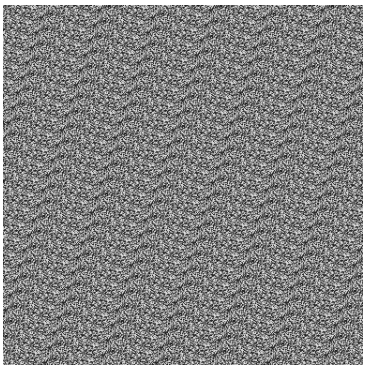
Figure 7.3: Cryptanalyze Girl.bmp using extracted K from f_m: CKBA

when $q = \alpha$; if $\alpha + \beta \bmod 4 = 0$, $f'(x,y) = f(x,y)$ when $q = \alpha + \beta$; and if $\alpha \bmod 4 = (\alpha + \beta) \bmod 4 = 0$, $f'(x,y) \equiv f(x,y)$. **C4**) All other gray values: If $\alpha \bmod 8 = 0$, $f'(x,y) = f(x,y)$ when $q = \alpha$; if $\alpha + \beta \bmod 8 = 0$, $f'(x,y) = f(x,y)$ when $q = \alpha + \beta$; and if $\alpha \bmod 8 = (\alpha + \beta) \bmod 8 = 0$, $f'(x,y) \equiv f(x,y)$.

2) For a sub-region in the plain-image with fixed gray value, at most eight* gray values will be contained in the corresponding sub-region of the cipher-image. Such a fact will make the edge of this sub-region appear in the cipher-image.

Apparently, if the cipher-image have many unchanged pixels and/or the plain-image have many sub-regions with fixed gray values, it will be possible to obtain some useful information about the plain-image by only observing the cipher-image. In Figure 7.5, we give the experimental result about a specially designed image, which contains pixels in all four classes (The gray values of the 16

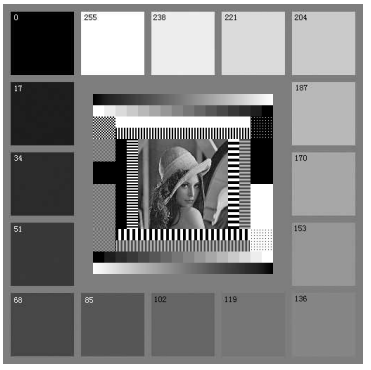
*The number is determined by the fixed gray value: C1 – 1, C2 – 1 or 2, C3 – $1 \sim 4$, C4 – $1 \sim 8$.



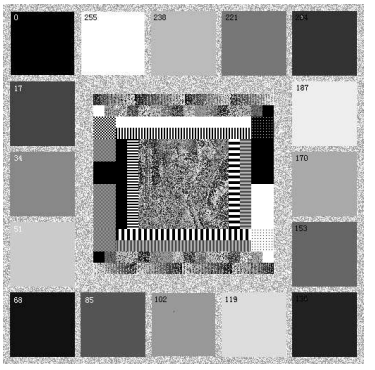
a) Mask image f'_m (384 × 384) generated from f_m (256 × 256)

b) Decrypted Girl.bmp by f'_m

Figure 7.4: Cryptanalyze Girl.bmp using f'_m generated from f_m: CKBA



a) Test_Pattern.bmp

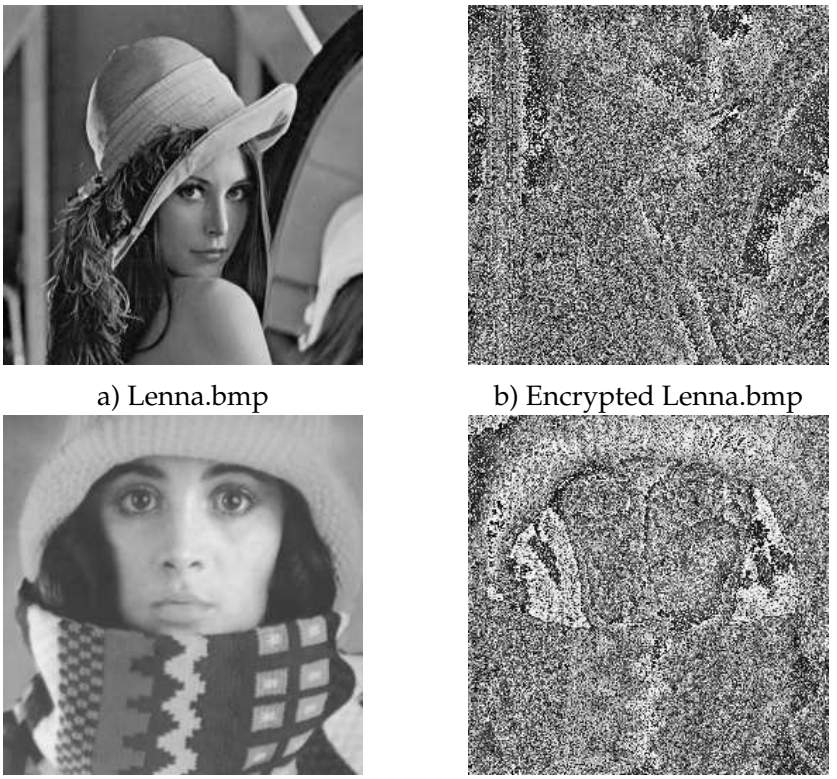


b) Encrypted Test_Pattern.bmp

Figure 7.5: A special image encrypted with BRIE

squares are respectively 0, 17, 34, ..., 221, 238, 255). The related parameters are $\alpha = 2, \beta = 4, x(0) = 0.75$ and the chaotic system is selected as Logistic map with control parameter 3.9.

In fact, the second fact can be extended to more general case. For a given sub-region, if all gray values are close and only a few LSBs of the values are different, there will be enough similar pixels in the sub-cipher-region to cause the edge to emerge in the cipher-image. Generally speaking, the larger the sub-region is and the closer the gray values are, the more clear the edge will be. In Figure 7.6, Lenna.bmp and Miss.bmp are shown as examples. In the cipher-images, we can find many important edges of the plain-images.



a) Lenna.bmp b) Encrypted Lenna.bmp

c) Miss.bmp d) Encrypted Miss.bmp

Figure 7.6: Lenna.bmp and Miss.bmp encrypted by BRIE, $\alpha = 5, \beta = 1, x(0) = 0.75$

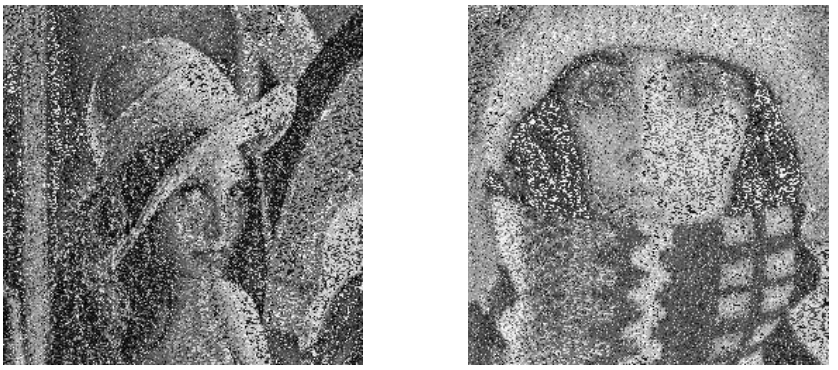
§7.4.2 Security Problem about α, β

The selection of α, β is not mentioned in [132]. We find α, β must yield the following three restrictions to avoid possible insecurity, the number of such values is only $7 \times 7 - 7 - 2 = 40$, which is dramatically small and will be useful for cryptanalysis.

R1) $\mathbf{1 \leq \alpha \leq 7, 1 \leq \beta \leq 7}$. Consider $ROLR_p^q = ROLR_p^{q+8}$, this restriction is natural.

R2) $\boldsymbol{\alpha + \beta \neq 8}$. If $\alpha + \beta = 8$, beyond half gray values will obey $f'(x, y) = f(x, y)$ (recall the discussion in the last subsection). Such a fact will cause the plain-image is roughly leaked from the cipher-image. In Figure 7.7, we give the results about Lenna.bmp and Miss.bmp when $\alpha = 6, \beta = 2, x(0) = 0.75$.

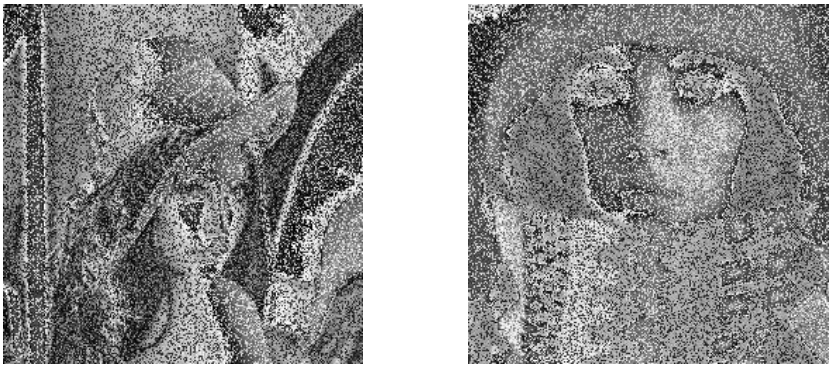
R3) $\boldsymbol{\alpha \bmod 8 \neq 1, 7}$ or $\boldsymbol{(\alpha + \beta) \bmod 8 \neq 1, 7}$. If the restriction is not satisfied (when $\alpha = 1, \beta = 6$ or $\alpha = 7, \beta = 2$), all plain-pixels will be encrypted by one-bit $ROLR$ operation since $ROLR_p^7 = ROLR_{1-p}^1$ and $ROLR_p^9 = ROLR_p^1$. Con-



a) Encrypted Lenna.bmp b) Encrypted Miss.bmp

Figure 7.7: Lenna.bmp and Miss.bmp encrypted by BRIE, $\alpha = 6, \beta = 2, x(0) = 0.75$ (compare them with Figure 7.6)

sequently, rather larger visual information of the plain-image will leak from the cipher-image. When $\alpha = 1, \beta = 6, x(0) = 0.75$, the results about Lenna.bmp and Miss.bmp are given in Figure 7.8. We can see the cipher-images contain so many strong edges that one eavesdropper can guess the plain-image.



a) Encrypted Lenna.bmp b) Encrypted Miss.bmp

Figure 7.8: Lenna.bmp and Miss.bmp encrypted by BRIE, $\alpha = 1, \beta = 6, x(0) = 0.75$ (compare them with Figure 7.6)

§7.4.3 Overestimated Security to Brute-Force Attack

In [132] J.-C. Yen and J.-I. Guo claimed that there are 2^{MN+1} possible encryption results since the cipher-image is determined by $\{b(i)\}_{i=0}^{MN}$; because all $\{b(i)\}$ keep secret to illegal users and the reconstruction of the chaotic orbit $\{x(i)\}_{i=0}^{\lceil (MN+1)/8 \rceil - 1}$ is rather difficult, BRIE is secure enough. However, the above

statement is not true because of the following fact: total $MN+1$ bits are uniquely determined by the chaotic system and its initial condition $x(0)$. Once one gets $x(0)$, he can easily reconstruct $\{b(i)\}_{i=0}^{MN}$ to decrypt the cipher-image. $x(0)$ can be determined by brute-force searching. Of course, to break BRIE, we also should know α, β besides $x(0)$.

Now let us calculate the total number of available secret keys. Assume the chaotic systems is iterated with floating-point arithmetic of double precision, then $x(0)$ will have 63 meaningful bits (the sign bit must be zero since $x(0) \geq 0$). Consider the number of available α, β is 40, the total number of keys is 40×2^{63}.

The exact computation complexity of the brute-force attack is estimated as follows. For each key, $\lceil (MN+1)/8 \rceil$ chaotic iterations are needed to generate $\{b(i)\}_{i=0}^{MN}$, and MN $ROLR$ operations are needed to decrypt the cipher-image. Assume one chaotic iteration and one $ROLR$ operation consume same time, the average attack complexity of BRIE to brute-force attack will be $(40 \times 2^{63}/2) \times 9(MN+1)/8 \approx 2^{67.5} \times MN$, which is much smaller than 2^{MN} when M, N are not too small. Assume $M = N = 512 = 2^9$, which is the typical size of a "large" digital image, the attack complexity will be only $2^{67.5} \times MN = 2^{85.5} \ll 2^{MN} = 2^{262144}$. Apparently, the security of BRIE is overestimated by J.-C. Yen and J.-I. Guo in [132], even under brute-force attack.

§7.5 Known/Chosen-Plaintext Attack to BRIE

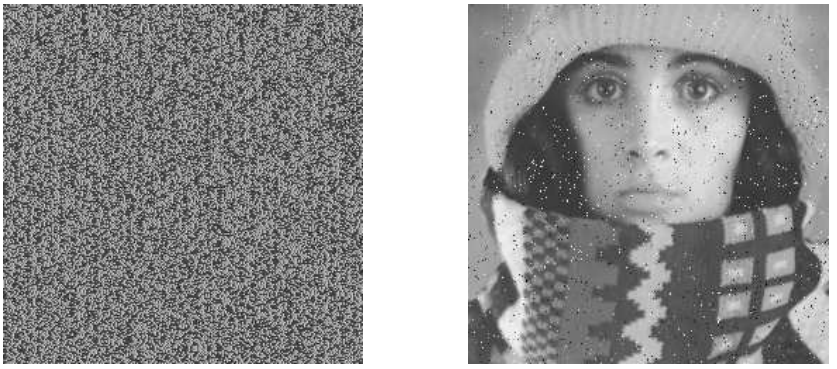
If one can get only one plain-image, he can break BRIE easily and fast, which corresponds to the known/chosen-plaintext attack in cryptanalysis.

§7.5.1 Breaking BRIE with Mask Array Q

Assume the known/chosen plain-image is f and its cipher-image is f' (both $M \times N$). For the plain-pixel $f(x,y)$, the cipher-pixel $f'(x,y)$ must be one of the 8 values: $ROLR_0^1(f(x,y)) \sim ROLR_0^7(f(x,y))$. By comparing $f(x,y)$ and $f'(x,y)$, we can easily find at least one integer* $q(x,y)$, which satisfies $f'(x,y) = ROLR_0^{q(x,y)}(f(x,y))$. Repeat this procedure, we can get a mask array $Q = [q(x,y)]_{M\times N}$. If $f(x,y)$ is a gray value in class C4 (recall §7.4.1), $q(x,y)$ can be used to decrypt any cipher-pixels at (x,y). If $f(x,y)$ is a gray value in class $C1 \sim C3$, generally $q(x,y)$ cannot be used to decrypt cipher-pixels at (x,y). Fortunately, for most digital images, the number of $C1 \sim C3$ pixels is much smaller than C4 pixels. Consequently, the mask array Q can be employed to decrypt other cipher-

*If $f(x,y)$ belongs to class C4, only unique such integer exists. For class C1, the number of such integers are 8; for C2, the number is 4; for C3, the number is 2.

images encrypted by BRIE with same keys. If the size of the cipher-images is not larger than the size of Q, the plain-images can be entirely recovered except a few plain-pixels. Select Lenna.bmp as the known/chosen plain-image, we obtain a mask array Q and then successfully cryptanalyze the cipher-image of Miss.bmp. The mask array Q and decrypted Miss.bmp are given in Figure 7.9, where Q is transformed to a pseudo-random image f_Q as follows: $F_Q(x,y) = q(x,y) \times 32$. We can see a few pixels in Miss.bmp cannot be correctly cryptanalyzed because of the corresponding pixels in Lenna.bmp are $C1 \sim C3$ pixels.



a) Mask array Q b) Decrypted Miss.bmp by Q

Figure 7.9: Cryptanalyze Miss.bmp using mask array Q generated from known/chosen Lenna.bmp, $\alpha = 5, \beta = 1, x(0) = 0.75$

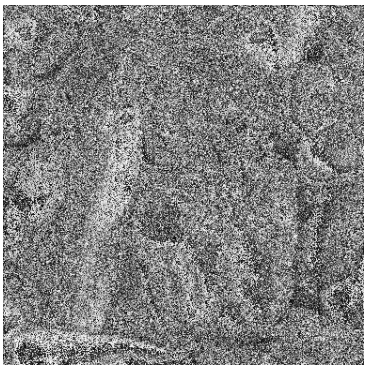
Using Q as the cryptanalytic tool has two problems: a) For a cipher-image whose size is larger than $M \times N$, only $M \times N$ pixels can be recovered. If the image is much larger than $M \times N$, the recovered part cannot reflect the whole scene of the plain-image. See Figure 7.10 for the cryptanalytic result of a larger image Peppers.bmp, whose size is 384×384 (larger than 256×256). b) If the known image contains too less C4 pixels, there will not be enough efficient $q(x,y)$ to decrypt cipher-pixels. The second problem can be overcome by increasing the number of known plain-images.

§7.5.2 Finding the Secret Keys from Q

Obviously, the best solution to the problems of Q is to get the secret keys of BRIE α, β and $x(0)$. Once Q is obtained, we can deduce α, β and equivalent $x(0)$ by the following steps.

Step 1: Divide the known/chosen plain-image into 8-pixel blocks, and find a C4

a) Peppers.bmp (384×384)



b) Encrypted Peppers.bmp

c) Decrypted Peppers.bmp by Q

Figure 7.10: Cryptanalyze Peppers.bmp using mask array Q generated from known/chosen Lenna.bmp, $\alpha = 5, \beta = 1, x(0) = 0.75$

pixel $f(x^*, y^*)$ followed by 2 consecutive C4 blocks* (generally it is easy for most images).

Step 2: Assume $\alpha' = 1 \sim 7$ and $\beta' = 1 \sim 7$.

Step 3: If α', β' disobey the restrictions **R1**, **R2** and **R3** described in Sect. §7.4.2, go to **Step 2**;

Step 4: Calculate the following four values: $q(1) = \alpha', q(2) = (\alpha' + \beta') \bmod 8, q(3) = 8 - q(1), q(4) = (8 - q(2)) \bmod 8$.

Step 5: Get 16 bits $\{b(1), \cdots, b(i), \cdots, b(16)\}$ from the mask values $q(x, y)$ corresponding to the 16 C4 plain-pixels starting from $f(x^*, y^*)$ as follows†:

- if $q(x, y) \notin \{q(1), q(2), q(3), q(4)\}$, go to **Step 2**;
- if $q(x, y) \in \{q(1), q(3)\}$, $b(i) = 0$;

*Here, "C4 block" means that all pixels in this block are C4 pixels.

†Note that $\{q(1), q(3)\} \cap \{q(2), q(4)\} = \varnothing$ for any α, β.

- if $q(x,y) \in \{q(2), q(4)\}$, $b(i) = 1$.

Step 6: Generate two binary decimals using $b(1) \sim b(16)$: $x1 = \sum_{i=1}^{8} b(i) \times 2^{-i}$ and $x2 = \sum_{i=1}^{8} b(i+8) \times 2^{-i}$.
Step 7: If $x2$ and $x1$ yield the equation of the employed chaotic system, mark the current α' and β' as a candidate for the right α and β. Go to **Step 2** until $\alpha' = 7$ and $\beta' = 7$.
Step 8: Search the right α and β in all marked candidates.
Step 9: Brute-forcedly search the other $n - 8$ bits of $x1$, where n is the meaningful bit number of the chaotic orbit, i.e., the finite realizing precision.

In the above procedure, if the 16 continuous C4 pixels are the first 16 pixels of plain-images, then $x1 = x(0)$; otherwise $x1$ is a equivalent key of $x(0)$ since $x1$ also can be used to generate chaotic sequence after $x1$. The search complexity of the above procedure is chiefly determined by **Step 9**. When $n = 63$ (double precision floating-point arithmetic), it is 2^{55}, which is still rather large. But compared with the complexity of simple brute-force attack (see §7.4.3), the key entropy decreases by at least $\log_2(40 \times 2^8) \approx 13.3$ bits.

§7.6 Can We Improve CKBA and BRIE?

In above sections, we have shown neither CKBA nor BRIE are secure enough to known/chosen plaintext attack, from both theoretical and experimental points of view. In this section, we will study some remedies to the two chaotic image encryption methods and discuss their performance of improving the security.

§7.6.1 Improving CKBA

The simplest idea to enhance CKBA is increasing the bit size (n) of $key1$ and $key2$, and the one (n') of $x(0)$. Accordingly, the basic criterion should be changed to $\sum_{i=0}^{7}(a_i \oplus d_i) = n/2$*. Such a simply enhanced CKBA will be stronger to ciphertext-only attack. Assume $n > 8$ and $n' > 16$, we can calculate the attack complexity is $(2^{n'-1}/(n'/2)) \times (2^n \times C_n^{n/2}/2) \times (MN)^2 = 2^{n+n'-1}/n' \times C_n^{n/2} \times (MN)^2$. When $n = n' = 32$ (consider the fact that 32-bit data is widely used in digital computers) and $M = N = 512 = 2^9$, the complexity will be approximately $2^{123.16}$. In addition, when $n' = 32$, the cycle length of $\{x(i)\}_{i=0}^{MN/8-1}$ will be large enough for almost all plain-images†, so it will be impossible to generate larger f'_m

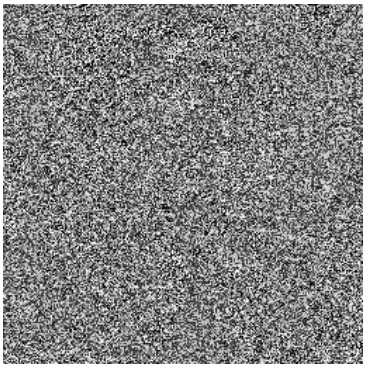
*The basic criterion can also be replaced with other ones, such as $\sum_{i=0}^{7}(a_i \oplus d_i) \in [n_1, n_2] \subseteq [1, n-1]$. Such a trivial modification can increase the attack complexity to ciphertext-only attack by some bits.

†Even for a "huge" image (4096×4096), MN is only $2^{24} \ll 2^{32}$.

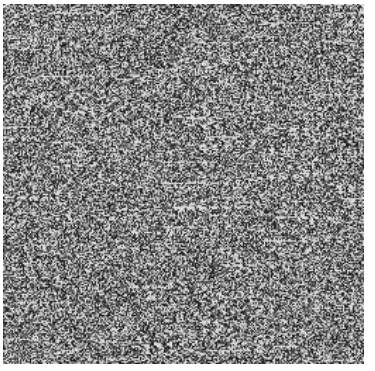
from a known f_m. However, it can not increase the complexity extracting K from f_m, since the complexity is just determined by M and N.

Another remedy is to add the control parameter(s) of the employed chaotic system as a secret sub-key. It can only enhance the capability against ciphertext-only attack, because different control parameters will make entirely different chaotic orbits even when the initial conditions are same. But it can not enhance the security to known-plaintext and chosen-plaintext attacks, either. Apparently, f_m can still be obtained without knowing the secret control parameter, and then the control parameter and the initial condition can be simultaneously extracted from the chaotic orbits.

Finally, let us discuss what the condition will be if some other advanced algorithms are employed to generate chaotic pseudo-random binary sequence $\{b(i)\}_{i=0}^{2MN-1}$. Apparently, they will make the extraction of K from f_m more difficult. But f_m is still available to decrypt the plain-image whose size is not much larger than the size of the known/chosen plain-image, and the complexity of ciphertext-only attack will not be influenced. To avoid the generation of larger f'_m from the known f_m, larger n' or the floating-point arithmetic is suggested being used to generate $\{x(i)\}_{i=0}^{MN/8-1}$. In Figure 7.11, we show the cipher-image of Lenna.bmp and the mask image under floating-point arithmetic. It can be seen that the mask image and the cipher-image are more disorderly than the ones given in Figure 7.1b and Figure 7.2a. However, the advanced algorithms to generate pseudo-random bit sequence and floating-point arithmetic need more computation complexity, so the enhanced CKBA will run slower than the original one.



a) Encrypted Lenna.bmp with floating-point arithmetic



b) Mask image with floating-point arithmetic

Figure 7.11: Using floating-point arithmetic in CKBA

To sum up, it is easy to enhance the security of CKBA to ciphertext-only at-

tack, but it is rather difficult to essentially enhance the security to known-plaintext and chosen-plaintext attacks. In fact, the essential reason of the above known-plaintext and chosen-plaintext attacks is the encryption procedure of CKBA (see Eq. (7.1)). But if we change the encryption procedure, CKBA will become an entirely different encryption scheme.

§7.6.2 Improving BRIE

To improve the security of BRIE to brute-force attack and the attack of getting the secret keys from Q, some simple modifications will be efficient, such as increasing the bit number of $x(0)$, adding control parameters of the chaotic system to the secret keys. But neither of them can improve the security to the known/chosen-plaintext attack with Q.

To escape from the known/chosen-plaintext attack based on Q, some complicated modifications should be made, such as cascading an extra cipher to perturb the cipher-image after BRIE[144], or using pseudo-randomly generated α and β by cipher-pixels and extra secret keys[22, 151]. Here, the security of the modified BRIE will be ensured by the new parts, not the BRIE itself.

§7.7 Conclusion

In this chapter, we point out two Yen-Guo's chaotic image encryption methods CKBA and BRIE proposed in [132, 134] are not secure. Known/chosen plaintext attacks can break them easily with only a pair plain-image and cipher-image. Some more security defects of BRIE are also found and discussed in detail.

Conceptually speaking, cryptanalyses given in this chapter can also be extended to break other Yen-Guo's image encryption methods. The lessons given by our cryptanalyses show that there exist a gap between signal/image encryption and cryptology, which should be bridged.

Part III

New Ways approach Digital Chaotic Ciphers

Chapter 8
CCS-PRBG Based Chaotic Stream Ciphers

Based on the comprehensive survey and cryptanalytic works on recently-proposed digital chaotic ciphers, some new approaches to design digital chaotic ciphers will be introduced in this chapter and the next chapter. Contents in the two chapters are based on the lessons learned from all known cryptanalyses on insecure digital chaotic ciphers and experience extracted from all secure ones at present. This chapter introduces a new idea on chaotic PRBG and its applications in stream-cipher cryptography, and the next chapter introduces a fast encryption chaotic cipher combing a chaotic stream sub-cipher and a chaotic block sub-cipher. Multiple chaotic systems are used in both chapters to enhance security of designed chaotic ciphers (recall my discussion in §2.6.1).

§8.1 Introduction

As we introduced in §2.2.1, a large number of stream cipher have been proposed based on chaotic PRNG-s. Because most chaotic PRNG-s used in chaotic stream ciphers only involve dynamics of a single chaotic system, there may exist potential insecurity caused by intelligent methods to extract useful information from chaotic orbits, such as the ones widely-used in cryptanalysis of chaos synchronization based secure communications[28–32, 35, 39, 42–44]. Also, the breaking of some chaotic PRNG-s based stream ciphers[24, 58, 60, 67] also emphasize such a threaten. We have suggested in §2.6.1 that using multiple chaotic systems in a digital ciphers is a general way to enhance its security against dynamical analyses based attacks. In fact, there are several chaotic stream ciphers using multiple chaotic systems[22, 45, 112, 119, 124], but some proposers were not aware of all advantages of multiple chaotic system in their ciphers.

In this chapter, we investigate the possibility to use only two (the least number of "multiple" chaotic systems) chaotic systems to realize better security against potential attacks and also reach better overall performance (taking encryption speed and implementation into considerations). A novel pseudo-random bit generator (PRBG) based on a couple of chaotic systems (called CCS-PRBG in short) is presented. Initial theoretical analyses and experimental data show that it has desired cryptographic properties and can be used to construct stream ciphers with high security. Generally speaking, we can regard CCS-PRBG as a nearly "perfect" nonlinear PRBG. When we design a new stream cipher, we can use it just like we use LFSR-s or NLFSR-s in conventional stream

ciphers[144, 145]. Of course, it is absolutely right that CCS-PRBG should have much higher security than LFSR-s and has equivalent (maybe also higher?) security to NLFSR-s. As applications of CCS-PRBG in stream-cipher cryptography, we introduce several typical designs of digital stream ciphers, which can reach a considerably good trade-off between its security and usability (high encryption speed and low implementation cost).

The organization of this chapter is as follows. In §8.2, CCS-PRBG and its digital realization with finite precision are introduced. Analyses on cryptographic properties of CCS-PRBG, including some experimental results, are given in §8.3. In §8.4, several examples of chaotic stream ciphers based on CCS-PRBG are suggested and discussion on their security is also given. The last section gives the conclusion and some open topics for future research.

§8.2 Couple Chaotic Systems Based PRBG (CCS-PRBG)

In above context of this dissertation, we have discussed that chaotic PRNG-s based on a single chaotic system are potentially insecure, since the output pseudo-random sequence can expose some information about the employed chaotic systems. In this section, we present a novel pseudo-random bit generator (PRBG) based on a couple of chaotic systems, which can provide higher security than other chaotic PRBG-s because **two** chaotic systems are employed to generate mixing pseudo-random bits. In this chapter we call it CCS-PRBG as a abbreviation of "Couple Chaotic Systems Based PRBG". The basic idea used in CCS-PRBG is to generate pseudo-random bits by comparing two different and asymptotically independent chaotic orbits, and it seems to be cryptographically strong to disable extracting information from the generated pseudo-random bits. Using CCS-PRBG like other PRBG in conventional stream-cipher cryptography, some chaotic stream ciphers can be designed, which will be discussed in §8.4.

§8.2.1 Definition

Assume there are two different one-dimensional chaotic maps $F_1(x_1, p_1)$ and $F_2(x_2, p_2)$: $x_1(i+1) = F_1(x_1(i), p_1)$, $x_2(i+1) = F_2(x_2(i), p_2)$, where p_1, p_2 are control parameters, $x_1(0), x_2(0)$ are initial conditions, and $\{x_1(i)\}$, $\{x_2(i)\}$ denote the two chaotic orbits.

Define a pseudo-random bit sequence as follows:

$$k(i) = g(x_1(i), x_2(i)) = \begin{cases} 1, & x_1(i) > x_2(i) \\ \text{null}, & x_1(i) = x_2(i) \\ 0, & x_1(i) < x_2(i) \end{cases}. \tag{8.1}$$

When some requirements are satisfied, the chaotic PRBG will have perfect cryptographic properties and can be called "a **C**ouple of **C**haotic **S**ystems based **P**seudo-**R**andom **B**it **G**enerator" (CCS-PRBG). These requirements are:

- *R1*) – $F_1(x_1, p_1)$ and $F_2(x_2, p_2)$ are surjective maps defined on a same interval $I = [a, b]$;
- *R2*) – $F_1(x_1, p_1)$ and $F_2(x_2, p_2)$ are ergodic on I, with unique invariant density functions $f_1(x)$ and $f_2(x)$;
- *R3*) – One of the following conditions holds: $f_1(x) = f_2(x) = f(x)$, or $f_1(x), f_2(x)$ are both even symmetrical to $x = (a + b)/2$;
- *R4*) – $\{x_1(i)\}, \{x_2(i)\}$ are asymptotically independent as $i \to \infty$.

If one of chaotic map is replaced by a constant $c \in I$, $k(i)$ will be simplified to the pseudo-random sequence in [61] and the chaotic threshold sequence in [151]. From such a viewpoint, CCS-PRBG can be regarded as the generalized version of them with "pseudo-random and time-variant threshold parameter": one chaotic orbit is binarized by anther chaotic orbit, the second chaotic orbit behaves like the threshold constant in [61, 151]. Also, we can consider CCS-PRBG as two inter-controlled chaotic PRBG-s, since $\{x_2(i)\}$ can be considered as the threshold sequence of $\{x_1(i)\}$ and vice versa.

§8.2.2 Digital Realization with Perturbation

When CCS-PRBG is realized in digital world, the perturbation-based algorithm in [81] is suggested improving dynamical degradation of digital chaotic systems contained in CCS-PRBG. The algorithm can be described as follows.

Use two simple PRNG-s to generate two pseudo-random signals*, which are used to perturb n_l lowest bits of $\{x_1(i)\}, \{x_2(i)\}$, with intervals Δ_1, Δ_2. The maximal length linear feedback shift registers (m-LFSR) are the best perturbing PRNG-s in hardware realizations, and the standard `rand()` function embedded

*Please see [81] for more details on how to generate the perturbing signals. Of course, we can use some other generation algorithms, the only requirement is that the generated signals should be pseudo-uniformly distributed in the definition domain of the perturbed chaotic system.

in almost all programming languages. Different from [81], here we suggest determining n_l as follows: $n_l \geq \lceil \lambda \cdot \log_2 e \rceil = \lceil 1.44\lambda \rceil$, where λ is Lyapunov exponent of the perturbed chaotic map. It is based on such a fact: when the finite computing precision is n (bits), the least difference (equals to 2^{-n}) between two signals will become $e^{\lambda} \cdot 2^{-n}$ after one iteration averagely (under fixed-point arithmetic). To keep the dynamical characteristics of the chaotic systems, $n_l \ll n$ should also be satisfied. Although the perturbing signal is much smaller than chaotic signal, it can still drive $\{x_1(i)\}, \{x_2(i)\}$ to very complex orbits since chaos is sensitive to initial conditions. The combination of digital chaos and pseudo-randomness of PRNG-s will make both chaos-theory-based and conventional cryptanalyses much more difficult.

Another trivial problem existing in digital CCS-PRBG is: when $x_1 = x_2$, $g(x_1, x_2)$ will not output pseudo-random bit, which is inconvenient in secure communications with fixed transmission rate since occasional null outputs can make the CCS-PRBG pause for a while. In such situations, an extra simple PRNG-3 can be introduced to determine $k(i)$. The digital CCS-PRBG with perturbation is shown in Figure 8.1. We can see that it can be easily realized by both hardware and software.

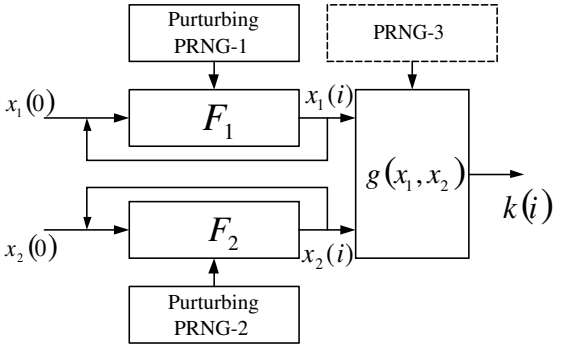


Figure 8.1: The digital realization of CCS-PRBG with perturbation

§8.3 Cryptographic Properties of Digital CCS-PRBG

For $\{k(i)\}$ generated by digital CCS-PRBG, the following cryptographic properties are satisfied: 1) balance on $\{0, 1\}$; 2) long cycle-length; 3) high linear complexity approximating to half of the cycle-length; 4) δ-like auto-correlation; 5) cross-correlation near to zero; 6) chaotic-system-free (see below for explanation). Detailed discussions are given as follows, with some experimental results.

§8.3.1 Balance

Theorem 8.1: *If two chaotic maps satisfy the above-mentioned requirements R1–R4, we can get* $P\{k(i)=0\}=P\{k(i)=1\}$, *i.e.,* $k(i)$ *is balanced on* $\{0,1\}$.

Proof: Because $F_1(x_1,p_1)$ and $F_2(x_2,p_2)$ are ergodic on $I=[a,b]$ (requirement *R2*), the orbits generated from almost all initial conditions will lead to the same distribution functions $f_1(x), f_2(x)$[23]. From requirement *R4*, the orbits $\{x_1(i)\}, \{x_2(i)\}$ are asymptotically independent, so the probabilities of $x_1 > x_2$ and $x_1 < x_2$ as $i \to \infty$ will be:

$$P\{x_1 > x_2\} = \int_a^b \int_a^x f_1(x) f_2(y)\, \mathrm{d}y\, \mathrm{d}x \tag{8.2}$$

$$P\{x_1 < x_2\} = \int_a^b \int_a^x f_2(x) f_1(y)\, \mathrm{d}y\, \mathrm{d}x \tag{8.3}$$

When requirement *R3* holds, we can prove $P\{x_1 > x_2\} = P\{x_1 < x_2\}$:
R3–1) $f_1(x) = f_2(x) = f(x)$:

$$P\{x_1 > x_2\} = P\{x_1 < x_2\} = \int_a^b \int_a^b f(x) f(y)\, \mathrm{d}y\, \mathrm{d}x. \tag{8.4}$$

R3–2) $f_1(x), f_2(x)$ are both even symmetrical to $x = (a+b)/2$:
Define the mirror orbits of x_1, x_2 as $x_1' = b - x_1, x_2' = b - x_2$. From the symmetry of $f_1(x), f_2(x)$, x_1', x_2' will have the same distribution $f_1(x), f_2(x)$, then we have:

$$P\{x_1 > x_2\} = P\{x_1' < x_2'\} = \int_a^b \int_a^{x'} f_2(x') f_1(y')\, \mathrm{d}y\, \mathrm{d}x = P\{x_1 < x_2\}. \tag{8.5}$$

Consider $x_1 > x_2 \to k(i) = 1$ and $x_1 < x_2 \to k(i) = 0$, $P\{x_1 > x_2\} = P\{x_1 < x_2\} \Rightarrow P\{k(i)=0\} = P\{k(i)=1\}$. The proof is complete. ■

Apparently, the above deduction is based on continuous conditions. When chaotic systems are discretely realized with perturbation, every chaotic orbit will be perturbed timely to a certain neighbor orbit by the small perturbing signal. Consequently, almost all orbits reach to the discrete versions of $f_1(x), f_2(x)$ with a little smoothing. For the discrete versions of $f_1(x), f_2(x)$, the above deduction also holds if $\int$ is replaced by $\sum$: Eq. (8.2) and (8.3) are replaced by

$$P\{x_1 > x_2\} = \sum_{x=a}^{b} \sum_{y=a}^{x} P_1\{x_1 = x\} \cdot P_2\{x_2 = y\} \tag{8.6}$$

and

$$P\{x_2 > x_1\} = \sum_{x=a}^{b} \sum_{y=a}^{x} P_2\{x_1 = x\} \cdot P_1\{x_2 = y\}. \tag{8.7}$$

From the approximate symmetry to $x = 1/2$ of x_1, x_2 when a digital CCS-PRBG is realized with perturbation, we can obtain the following result $P\{x_1 > x_2\} \approx P\{x_1 < x_2\}$. Therefore, the balance will be approximately preserved in the digital CCS-PRBG with perturbation.

§8.3.2 Long Cycle Length of the Pseudo-Random Bit Sequence

Without loss of generality, assume two m-LFSR-s are used as the perturbing PRNG-s, whose degrees are L_1, L_2, and perturbing intervals are Δ_1, Δ_2 respectively. Then the cycle length of $\{x_1(i)\}, \{x_2(i)\}$ are $\sigma_1\Delta_1(2^{L_1}-1), \sigma_2\Delta_2(2^{L_2}-1)$, where σ_1, σ_2 are two positive integers[81]. Thus, the cycle length of the bit sequence $\{k(i)\}$ will be:

$$\mathrm{lcm}(\sigma_1\Delta_1(2^{L_1}-1), \sigma_2\Delta_2(2^{L_2}-1)). \tag{8.8}$$

When Δ_1, Δ_2 and L_1, L_2 are selected to satisfy $\gcd(\Delta_1, \Delta_2) = 1$ and $\gcd(2^{L_1}-1, 2^{L_2}-1) = 1$, the cycle-length of $\{k(i)\}$ will be:

$$\mathrm{lcm}(\sigma_1, \sigma_2) \cdot \Delta_1\Delta_2(2^{L_1}-1)(2^{L_2}-1) \approx \mathrm{lcm}(\sigma_1, \sigma_2) \cdot \Delta_1\Delta_2 2^{L_1+L_2}. \tag{8.9}$$

Such a cycle length is long enough for most secure applications. Furthermore, there are still some methods that can be used to further prolong the cycle length, such as the one in [82].

§8.3.3 High Linear Complexity and Good Correlation Properties

Actually, the requirement *R4* and the balance of $\{k(i)\}$ imply that $\{k(i)\}$ is an i.i.d. (independent and identically distributed) bit sequence as $i \to \infty$. Therefore, it will have δ-like auto-correlation and near-to-zero cross-correlation. What's more, it has been proved (see [227]) that i.i.d. binary sequence has half-length linear complexity, so $\{k(i)\}_{i=1}^{n}$ will also have high linear complexity approximating to $n/2$ *. So let us discuss under what condition requirement *R4* will be satisfied for digital CCS-PRBG.

For any chaotic map, even if the initial condition or the control parameter has a very small difference, its orbit will become entirely different after limited iterations. If there is some initial information about the orbit, the information will decrease to zero as $i \to \infty$. The relation between two chaotic orbits can be considered as such information. In chaos theory, Kolmogorov entropy is defined to measure the decreasing rate of the information. For one-dimensional chaotic maps, Kolmogorov entropy is equal to Lyapunov exponent[206, 209]. If the initially known

*The cycle-length of $\{k(i)\}$ is $L = \mathrm{lcm}(\sigma_1\Delta_1(2^{L_1}-1), \sigma_2\Delta_2(2^{L_2}-1))$, not infinity. Hence, the linear complexity of $\{k(i)\}_{i=1}^{\infty}$ should be about $L/2$, not infinity either.

information is H, it will lose completely after $\eta \approx H/\lambda$ iterations[61], where λ is Lyapunov exponent. When chaotic systems are realized discretely, the information will decrease even faster since the quantization errors and small perturbing signals makes two orbits depart faster. So we can see, as long as there is initial difference between two chaotic orbits, they will become asymptotically independent as $i \to \infty$. Therefore, the equivalent requirement of *R4* is $\{x_1(i)\} \neq \{x_2(i)\}$, that is to say, $F_1 \neq F_2$, or $x_1(0) \neq x_2(0)$, or $p_1 \neq p_2$.

Because the independence of $\{x_1(i)\}, \{x_2(i)\}$ holds after η iterations, we suggest discarding the first m bits of $\{k(i)\}$, where $m > \eta$. It means m pre-iterations for the two chaotic maps should be done before $\{k(i)\}$ is output. Since m is not very large, such pre-iterations need only a little extra computation load.

Although analyses given here are entirely theoretic and qualitative, experiments strongly support the theoretical results (see the following Figure 8.2 and §8.3.5 for more details). In the future research, we will try to find the strict proof of $\{k(i)\}$ generated by CCS-PRBG is i.i.d. binary sequence*.

§8.3.4 Chaotic-System-Free Property

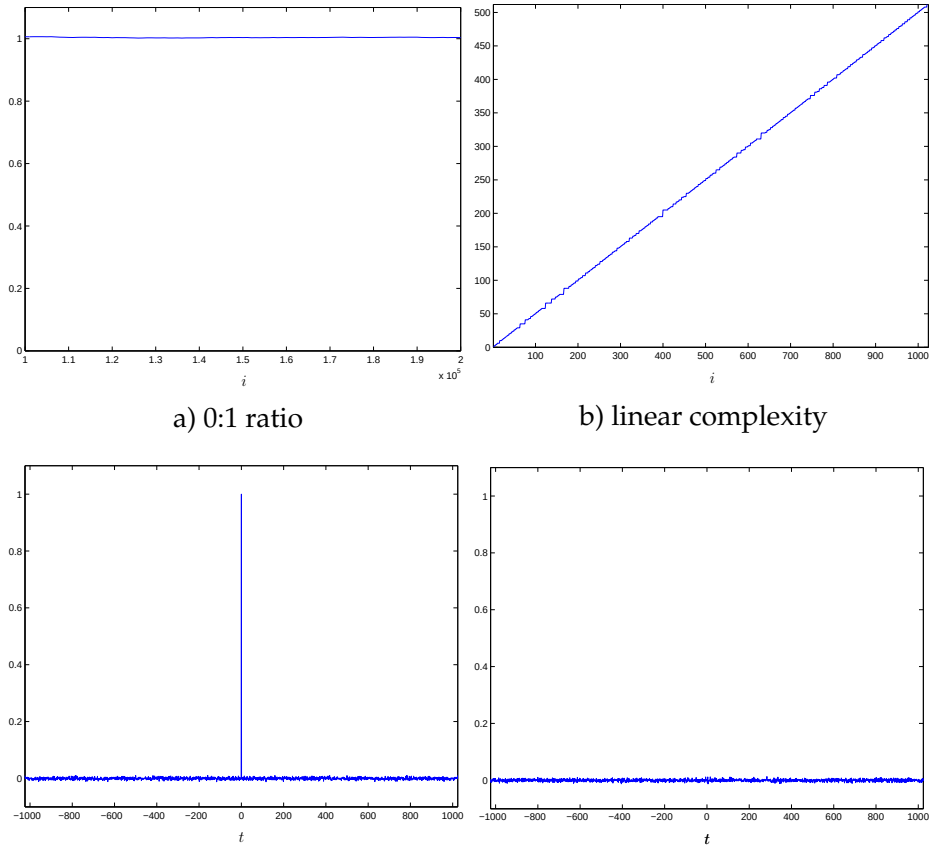
Consider there are many different chaotic maps satisfy the requirements *R1* and *R2*, and the requirement *R3* and *R4* just restrict the relation between the two chaotic systems, we call CCS-PRBG *chaotic-system-free*, which is a term to emphasize its wide use for a larger number of chaotic systems. Recall we discussed in §2.6.1, such a property should be optimal case for digital chaotic ciphers. Since PWLCM-s satisfy the requirements *R1–R4*, they are strongly suggested again.

§8.3.5 Experimental Results

In order to verify the theoretical results on cryptographic properties of digital CCS-PRBG with perturbation, some experiments are made. The two chaotic maps are both selected as the PWLCM (2.1). The finite computing precision is $n = 32$ (bits). The perturbing PRNG-s are selected as two m-LFSR-s, whose degrees are $L_1 = 16$, $L_2 = 17$ and whose perturbing intervals are $\Delta_1 = 99$, $\Delta_2 = 101$. The number of pre-iteration m is 16. Both initial conditions and control parameters are generated randomly, and a large number of sub-sequences of $k(i)$ are extracted from random positions to test the cryptographic properties. The 0:1 ratio, linear complexity and auto-correlation of one sub-sequence are shown in Figure 8.2a–c respectively. In Figure 8.2d, the cross-correlation of two sub-sequences with

*Of course, because the two involved pseudo chaotic orbits are actually deterministic, here i.i.d feature holds only in an approximate sense.

identical initial conditions but slightly different (2^{-n}) control parameters is given. We can see the experimental results coincide well with the theoretical analyses.



a) 0:1 ratio b) linear complexity

c) auto-correlation d) cross-correlation

Figure 8.2: Cryptographic properties of digital CCS-PRBG

§8.4 Construct Stream Ciphers Using Digital CCS-PRBG

Based on digital CCS-PRBG, many different practical stream ciphers can be constructed. We will see these stream ciphers can provide feasible solutions to the problems existing in other digital chaotic ciphers. Using different configurations of CCS-PRBG, many stream ciphers can be obtained conveniently with considerably low cost and simple realization, but without loss of security. Here, digital CCS-PRBG replaces the kernel role of LFSR in conventional stream-cipher cryptography.

§8.4.1 Some Examples of Stream Ciphers

• Cipher 1

Give a digital CCS-PRBG with perturbation, initial conditions $x_1(0), x_2(0)$ and control parameters p_1, p_2 are the secure key. $\{k(i)\}$ is directly used to encrypt (generally XOR) plaintext and decrypt ciphertext.

Apparently, this cipher is the simplest stream cipher based on digital CCS-PRBG. If finite computing precision is n (bits), the key entropy will be $4n$. Moreover, it is easy to be realized by hardware or software with rather low cost. On a 800MHz Pentium III PC, a software version based on the PWLCM (2.1) is developed with Turbo C 2.0 for test. The actual encryption speed reaches 9 Mbps under fixed-point arithmetic. Such a speed is faster than many other chaotic ciphers and can be acceptable in many secure applications. Under hardware realization, the speed will be promoted much and can be approximately estimated as follows: assume clock frequency is s MHz and the finite precision is n-bit, the estimated speed will be about $\frac{s}{n}$ Mbps (each n-bit digital division consumes about n clock cycles).

If some simple modifications are made on Cipher 1, enhanced stream ciphers with larger key entropy (higher security?) and faster speed can be obtained with a little extra complexity and cost. Two examples are given as follows to show how to extend Cipher 1.

• Cipher 2

Give four one-dimensional chaotic systems $CS_0 \sim CS_3$, and five m-LFSR-s $m\text{-}LFSR_0 \sim m\text{-}LFSR_4$, in which $m\text{-}LFSR_0 \sim m\text{-}LFSR_3$ are used to perturb $CS_0 \sim CS_3$. Before each iteration of $CS_0 \sim CS_3$, firstly use $m\text{-}LFSR_4$ to generate two 2-bits pseudo-random numbers $pn1(i)$ and $pn2(i)$. If $pn2(i) = pn1(i)$, do $pn2(i) = pn1(i) \oplus 1$; else do nothing. Then select $CS_{pn1(i)}$ and $CS_{pn2(i)}$ to compose the digital CCS-PRBG to generate $k(i)$. The secure key contains the initial conditions and control parameters of the four chaotic systems.

The key entropy will be $8n$ under n (bits) computing precision. $m\text{-}LFSR_4$ adds more complexity to the cryptanalysis so such a cipher is securer, with only double cost of realization and approximate encryption speed to cipher 1.

• Cipher 3

For some chaotic maps defined on $I = [0, 1]$, such as the PWLCM (2.1), the invariant density function is $f(x) = 1$. When they are realized in digital computers, every bit of the orbit will be balanced on $\{0, 1\}$. Based on such a fact, we can

define a generalized version of digital CCS-PRBG. Here assume finite computing precision is n (bits). For one iteration of $F_1(x_1, p_1)$ and $F_2(x_2, p_2)$, generate n bits $K(i) = k_0(i) \dots k_{n-1}(i)$ as follows:

for $j = 0$ to $n - 1$ do
$x_1(i, j) = x_1(i) \gg j$
$x_2(i, j) = x_2(i) \ll j$
$k_j(i) = g(x_1(i, j), x_2(i, j))$
end

Where » and « respectively denotes circular right shift operation and circular left shift operation. Apparently, a stream cipher based on generalized CCS-PRBG will run nearly n times faster than the one based on common CCS-PRBG, without loss of high security. When Cipher 3 is realized by hardware with parallel arithmetic technique, its encryption speed will be close to s Mbps when the clock frequency is s MHz [*]. Such a speed approximately equals to the speed of many conventional stream ciphers based on LFSR-s, such as Geffe generator and clock-controlled generator, and faster than some complicated stream ciphers[144, 145]. If we combine Cipher 2 and Cipher 3, both the security and the encryption speed can be improved much. Actually, in order to further enhance the security of Cipher 3, we can introduce another m-$LFSR_5$ to pseudo-randomly control the direction of the circular shift operation of x_1 and x_2.

In Table 8.1, we give a brief comparison of the above three ciphers and the combined cipher of Cipher 2 and Cipher 3. LFSR based ciphers are also listed as references. In this table n is the finite precision and a means the implementation cost of Cipher 1. Please note that LFSR based ciphers are generally insecure although its implementation cost is smaller. We can see Cipher 3 may be a promising part to design stream ciphers with desired overall performance. Also, compared with other chaotic stream ciphers, I believe we can use CCS-PRBG to construct new stream ciphers with better overall performance. Another possible use of CCS-PRBG is to construct product cipher with other cryptographical techniques. For example, CCS-PRBG can be used in CVES (which will be discussed in Chap. 9) to enhance its security.

§8.4.2 Security

Generally speaking, the security of the above ciphers can be ensured by cryptographic properties of digital CCS-PRBG discussed in §8.3. But we have known that many chaotic ciphers are not secure although they have some "good" statis-

[*] Apparently, the speed is chiefly determined by the fixed-point divisions needed in chaotic iterations. Since a n-bit digital divider consumes about n clock cycles for one n-bit division, the encryption speed of Cipher 3 will be close to $\frac{s}{n} \cdot n = s$ Mbps.

Table 8.1: A comparison of CCS-PRBG based stream ciphers

	Key Entropy	Speed (Hardware)	Implementation
Cipher 1	$4n$	$\frac{s}{n}$ Mbps	a
Cipher 2	$8n$	$\frac{s}{n}$ Mbps	$2a$
Cipher 3	$4n$	s Mbps	a
Cipher 2+3	$8n$	s Mbps	$2a$
LFSR based ciphers	/	s Mbps	$< a$

tical properties. So we should still investigate whether or not the ciphers based on digital CCS-PRBG is secure enough to known cryptanalysis methods.

At first, let us consider cryptanalyses of chaos synchronization based secure communications[28–32, 35, 39, 42–44]. They can work because chaos synchronization makes it possible to extract dynamical information of the chaotic systems. Since the transmitted signal must be used to realize synchronization of the transmitter and receiver, such information may be useful to restore the chaotic orbit and then extract the hidden message. For digital CCS-PRBG, because chaos synchronization is not used and two different chaotic orbits are employed to make pseudo-random keystream $k(i)$, the dynamics of the two chaotic systems cannot be obtained from the ciphertext. In addition, the pseudo-random perturbation also makes the cryptanalysis more difficult. Even if the plaintext is known, it is impossible to extract the two chaotic orbits just from $k(i)$. Hence, those methods, which are available to break secure communication approaches based on chaos synchronization, cannot be used to break the ciphers based on digital CCS-PRBG.

Other known cryptanalytic methods aim at specific weaknesses of concerned chaotic ciphers. The one in [59, 64] is available because of the degraded statistical properties of discrete chaotic systems, which has been considered carefully and been avoided by perturbation-based algorithm in digital CCS-PRBG. The one in [70] is based on a specific weakness of 2-D Hénon map and cannot be generalized to other chaotic systems. The ones in [66, 88, 97] can work well for the special weaknesses in the corresponding ciphers and also cannot be extended to break CCS-PRBG based ciphers with entirely different encryption structure.

Now we can see the ciphers based on digital CCS-PRBG are secure to all known cryptanalyses of digital chaotic ciphers. Of course, before we can finally say "digital CCS-PRBG based ciphers are secure enough", further cryptanalytic works of digital CCS-PRBG should be done. But the above discussion implies that digital CCS-PRBG may be a new promising candidate to construct stream ciphers with high security and low cost.

There is one notable defect in digital CCS-PRBG that should be mentioned

here. Assume $x_1(0) = x_2(0)$, when the control parameters are p_1, p_2, the generated pseudo-random bit sequence is $k(i)$; exchange the control parameters of the two chaotic maps, the generated pseudo-random bit sequence is $k'(i)$. If the system equation of two chaotic maps are identical, and they are perturbed with identical perturbing PRNG-s and identical perturbing intervals ($\Delta_1 = \Delta_2$), it is obvious that $k'(i) = \overline{k(i)}$, which is the natural result of $g(x_2, x_1) = \overline{g(x_1, x_2)}$. Such an effect will cause the key space size of the ciphers decrease $1/2$. To avoid this defect, different perturbing PRNG-s or perturbing intervals should be used, and $m > \max(\Delta_1, \Delta_2)$ is suggested. Also, using two chaotic systems with different system equations can solve this defect, such as the PWLCM (2.1) and skew tent map (2.3).

§8.5 Conclusion

In this chapter a novel chaotic PRBG based on a couple of chaotic systems (called CCS-PRBG) is proposed to construct new digital chaotic stream ciphers. Both theoretical and experimental analyses show that digital CCS-PRBG has desired cryptographic properties. The digital CCS-PRBG can be a kernel part in the design of new stream ciphers to replace LFSR-s' roles in conventional cryptography.

For CCS-PRBG, there are some open topics in future research:

- As we mentioned in §8.3.3, the strict proof of $\{k(i)\}$ is i.i.d. sequence is still a unsolved problem from a strict point of view.
- Some details on (hardware and software) implementation of CCS-PRBG based stream ciphers will be concerned.
- Possible cryptanalysis methods of the digital CCS-PRBG will be another open topic.

Chapter 9
A Novel Chaotic Encryption Scheme with very Fast Speed

§9.1 Introduction

In the digital world nowadays, the security of digital images/videos becomes more and more important since the communications of digital products over network occur more and more frequently. In addition, special and reliable security in storage and transmission of digital images/videos is needed in many digital applications, such as pay-TV, confidential video conferencing and medical imaging systems, etc. Generally speaking, the well-developed modern cryptography should be the perfect solution to this task. As we know, many perfect ciphers have been established and applied widely since 1970s, such as DES, IDEA and RSA[144, 145]. But many conventional ciphers cannot be directly used to encrypt digital video in real-time systems because their encryption speed is not fast enough, especially when they are realized by software. In addition, the existence of different compression algorithms in digital video systems makes it more complicated to incorporate the encryption component into the whole system. Thus, to protect the content of real-time videos, some specially-designed encryption methods are needed.

Recently, many specific video encryption schemes have been proposed[221, 228–237]. Most of them are joint compression-encryption methods, which are specially designed to provide reliable security for MPEG video stream[229, 230, 232–237]. From the works in [238–240], some video encryption schemes have been known to be not secure enough from strict cryptographic viewpoint. Actually, there still exist trade-offs between the security and the encryption speed in many video encryption systems[238].

This chapter considers the following question: is it possible to use digital chaos to design fast encryption schemes to solve problems with real-time video encryption? From our survey on digital chaotic ciphers in Chap. 2, we have known most chaotic ciphers run with rather slow speed. This fact makes it difficult to persuade crypto-practitioners to accept digital chaotic ciphers as candidates for actual applications. I have introduced CCS-PRBG base stream cipher to relax this embarrassment to some extent (Cipher 3 can run with high speed in hardware). But stream ciphers have essential defects: the secret key cannot be reused to avoid known/chosen plaintext attacks. In this chapter, a novel idea

using digital chaos to construct ciphers with very fast speed will be investigated.

Following the proposed idea (which will be introduced in the next section), a chaotic video encryption scheme (CVES in short) is carefully developed to fulfill demands of real-time video encryption. Initial analyses show it seems to be secure and can run with rather fast encryption speed, and can be realized simply by both hardware and software. CVES is independent of any video compression algorithm so it will not be limited by the format of encrypted video, which is one important merit of CVES compared with other video encryption systems. In addition, a generalized version of CVES is also developed to support random retrieval of encrypted video with considerable maximal time-out.

As the last chapter of main body of this thesis, this chapter should not be considered a complete solution to design difficulties of digital chaotic ciphers, but an active attempt to find general structure and design principles. Maybe my proposals will be found insecure soon, but I believe such an attempt is helpful to distinguish more facts on "what we should do" and "what we should not do". Our cryptanalytic experience implies that it is even more difficult to design a really secure digital chaotic cipher than to break it. In fact, the original CVES proposed in [112] has been found not secure enough. Although it is enhanced in this chapter, now it is too early to say the enhanced version is secure.

This chapter is organized as follows. A basic description on the novel idea used in CVES is given in §9.2. CVES and its extended version RRS-CVES (Random-Retrieval-Supported CVES) are detailedly described in §9.3. The performance of CVES/RRS-CVES is estimated in §9.4, respectively from the viewpoints of the encryption speed, security, realization and experiments. The last section is the conclusion.

§9.2 A Conceptual Description of the Proposed Idea

The kernel of the proposed idea using chaos to realize fast encryption is to combine a simple chaotic stream cipher and a simple chaotic block cipher (with time-variant S-boxes) to realize a much more complex product cipher. As we have discussed in Chap. 2, the use of multiple iterations is the main reason to make chaotic ciphers slow, but less iterations may bring security problems. Thus, we want to make single iteration possible by combining a stream cipher and a block cipher to ensure security. The design of CVES shows such an idea really works.

This idea can be described as follows: assume P_i, C_i respectively represents the i^{th} plaintext and the i^{th} ciphertext (both with n-bit formats), the encryption procedure is defined by

$$C_i = f_S\left(P_i \oplus x_i, i\right), \tag{9.1}$$

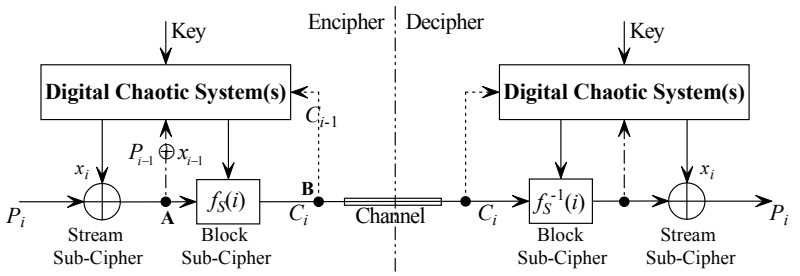


Figure 9.1: A Novel Idea Using Chaos to Construct Ciphers

where $f_S(\cdot, i)$ is a time-variant $n \times n$ S-box (a bijection defined on $\{0, 1, \cdots, 2^n - 1\}$) and x_i is extracted from the state(s) of a(many) chaotic system(s). Here, f_S is also pseudo-randomly controlled by the chaotic system(s). The secret key is selected as the initial condition(s) and the control parameter(s) of the employed chaotic system(s). Please see Figure 9.1 for its encryption/decryption procedure. To increase the complexity of the obtained cipher against possible attacks, internal feedback ($P_{i-1} \oplus x_{i-1}$ at point **A**) and ciphertext feedback (C_{i-1} at point **B**) should be added. Such a cipher can be considered as a combination of a simple stream cipher and a simple block cipher.

Here, let us see why internal feedback and/or ciphertext feedback are needed. Without any feedback, the above cipher will become the following collapsed version:

$$C_i = f'_S(P_i, i), \tag{9.2}$$

where f'_S is fixed for each position i. Such a cipher is actually a stream cipher using time-variant function $f'_S(i)$ instead of XOR to mask the plaintexts (see Figure 9.2). Although it has better security than stream cipher with XOR masking functions, a chosen plaintext attack can still break it and get all $f'_S(\cdot, i)$ with 2^n plaintexts: $0 \cdots 0 \cdots, 1 \cdots 1 \cdots, \cdots, (2^n - 1) \cdots (2^n - 1) \cdots$. When 2^n is not large enough, this chosen-plaintext attack works well. However, if 2^n is too large, then it will be not easy (or even practically impossible) to generate time-variant f_S with fast encryption speed. In fact, in CVES $n = 8$, which is too small to resist the above chosen plaintext attack.

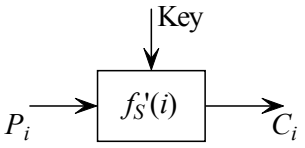
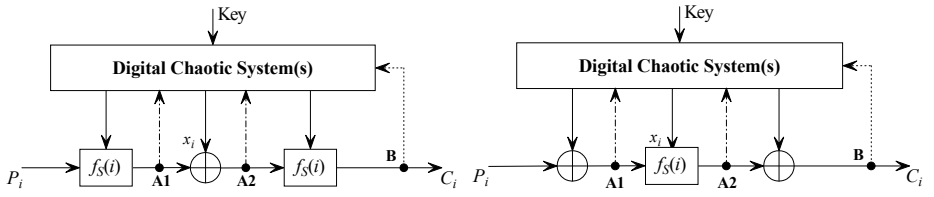


Figure 9.2: A Cipher without any Feedback in Figure 9.1

As a solution to this paradox, feedback is introduced to make f'_S also dependent on N_p previous plaintexts and then disable the above chosen plaintext attack. The strength against the above chosen plaintext (ciphertext) attack will be increased to $2^{n \cdot N_p}$. Of course, like in CBC mode, there will exist error propagation, and the propagation length is $N_p \cdot n$ bits. For video applications, N_p can be relatively large.

There is a nontrivial problem on the feedback. For the first plaintext, f'_S is still fixed since no previous plaintexts are available. An initial vector (IV) will be introduced to solve this issue: for each plain-message, the beginning $N_p \cdot n$ bits are randomly generated and serve as IV to encrypt/decrypt the first meaningful plaintext. As long as $2^{n \cdot N_p}$ is large enough, then it will be probabilistically difficult for an attacker to carry out cryptanalyses.

Although we have find some evidence to show security of the above cipher, it is still possible to find some attacks in future. Then the following two extended models shown in Figure 9.3 will be considered as possible solutions (the second XOR in Model 2 can be replaced with other functions, such as $x + b \bmod 2^n$), at present no any investigation has been made on them. Please note that actually digital chaotic systems are not requisites for the ciphers shown in this section, any cryptographical primitives can be used to design new ciphers. In future we will also study such possibility to generalize the idea to conventional cryptography.



a) Extended Model 1 b) Extended Model 2

Figure 9.3: Two extended models of the cipher in Figure 9.1

§9.3 Chaotic Video Encryption Scheme – CVES

The Chaotic Video Encryption Scheme (CVES) is shown in Figure 9.4. It is an enhanced version of the original CVES proposed in our paper [112]*. The plain-video is encrypted *cluster* by *cluster*, where a *cluster* can be one or more video frames.

In fact, we can also consider the video stream as a continuous bit-stream without any video format and take fixed-size bits as a *cluster*. Such an encryp-

*The original CVES in [112] is not secure because feedback is not used.

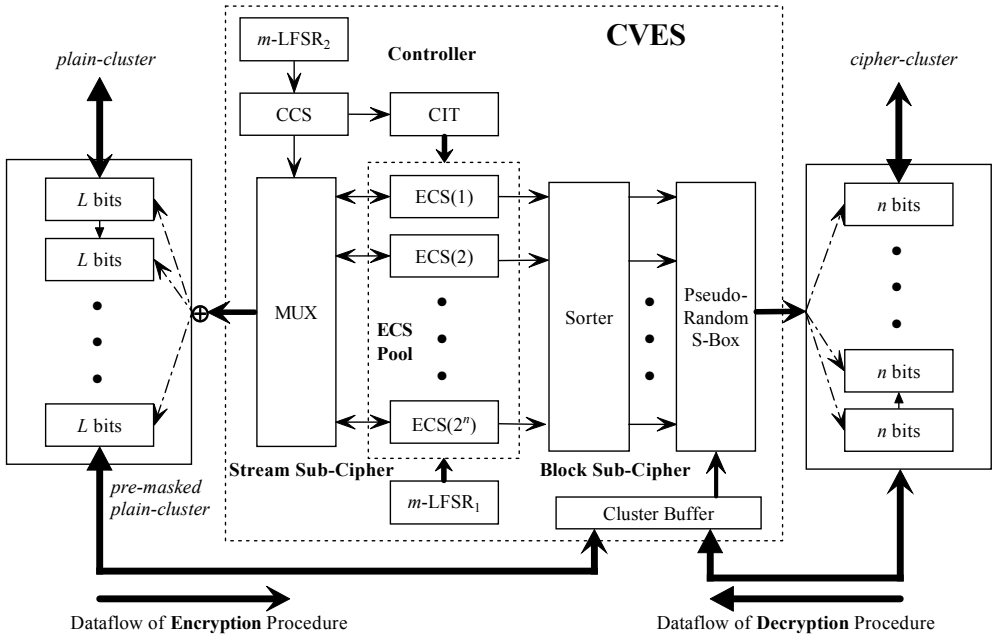


Figure 9.4: Encryption and Decryption Procedure of CVES

tion feature makes CVES independent of video format, and so no any negative influence about video format is introduced (such as data expansion of video data in some encryption-before-compression schemes). Apparently, we can combine CVES with useful ideas in other video encryption methods to obtain better overall performance. For example, we can employs the idea of "partial encryption"[221] to enhance CVES: only partial data in the whole video are encrypted with CVES so that the final encryption speed will be promoted much. Similarly, we can also use CVES as follows: only key frames (such as I-pictures in MPEG video[241]) are encrypted, and other frames (such as P-pictures and B-pictures in MPEG video) are simply skipped without encryption.

§9.3.1 Components

Before describing the encryption/decryption procedure of CVES, we firstly introduce the components of CVES.

1) **ECS Pool**: 2^n digital chaotic systems, which are called Encryption Chaotic Systems (ECS) and denoted by ECS$(1) \sim$ ECS(2^n), compose the kernel part of CVES – ECS Pool. All 2^n ECS-es are based on a same one-dimensional chaotic maps $F_e(x_e, p_e)$ defined on $I = [0,1]$, with different control parameters $p_e(1) \sim p_e(2^n)$. All ECS-es are realized in finite computing precision L (bits) with

perturbation-based algorithm, and one maximal length LFSR m-LFSR$_1$ is used as the perturbing PRNG. The degree of m-LFSR$_1$ is L_1, and the perturbing intervals of the 2^n ECS-es are $\Delta_e(1) \sim \Delta_e(2^n)$. The current states of the 2^n ECS-es $x_e(1) \sim x_e(2^n)$ are stored in 2^n L-bit memory units.

2) **CCS**: A single digital chaotic systems is used to control the initialization and the chaotic iterations of the 2^n ECS-es. It is called Control Chaotic System (CCS). CCS is also based on a one-dimensional chaotic map $F_c(x_c, p_c)$ defined on $I = [0, 1]$, which can be different from F_e. CCS is also realized in finite precision L (bits) with perturbation-based algorithm, and another maximal length LFSR m-LFSR$_2$ is used as the perturbing PRNG. The degree of m-LFSR$_2$ is L_2, and the perturbing interval of CCS is Δ_c.

3) **CIT**: A Control Information Table (CIT) is used to store the required information in CVES. In regards to the information stored in CIT, please see §9.3.2 and §9.3.3. The CCS and CIT compose the controller part.

4) **Stream Sub-Cipher**: A $2^n \times 1$ MUX controlled by CCS is employed to select an ECS to generate a L-bit chaotic key, which is used to XOR the *plain-cluster* L-bit block by L-bit block. The *plain-cluster* encrypted by the stream sub-cipher is called *pre-masked plain-cluster*.

5) **Block Sub-Cipher**: A $2^n \times 2^n$ L-bit sorter and 2^n n-bit memory units $S[0] \sim S[2^n - 1]$ compose a Pseudo-Random S-Boxes Generator (PRSBG). The generated pseudo-random $n \times n$ S-box is used to substitute the *pre-masked plain-cluster* n-bit block by n-bit block. Here please note that the S-boxes at encryption end and the ones at decryption end are inverse.

6) **Cluster Buffer**: A memory buffer to store pre-masked *cluster* and NC_{max} L-bit internal variables for internal feedback: $N_F(1) \sim N_F(NC_{max})$, which are used to pseudo-randomly perturb the generated S-box $S[0] \sim S[2^n - 1]$ when each pre-masked plaintext in current cluster is encrypted by the block sub-cipher. To make the first n-bit plain-block of next plain-cluster also dependent on plaintext in the last plain-cluster, we extend the number of stored pre-masked plain-blocks to be $NC_{max} + N_p$, where N_p more units are added to store N_p latest plain-blocks in last plain-cluster. As a whole, the cluster buffer is composed of two parts: $N_F(1) \sim N_F(NC_{max})$, and $N_F(-(N_p - 1)) \sim N_F(0)$.

§9.3.2 Encryption/Decryption Procedure

Based on the introduction to the components of CVES, we can describe the encryption procedure as follow. Here, we consider the $x_e(1) \sim x_e(2^n)$, $p_e(1) \sim p_e(2^n)$ and x_c, p_c as L-bit binary integers, not the binary decimals in $[0, 1]$ (under L-bit fixed-point arithmetic), to simplify the description.

- **Secret key**: $K = \{x_c, p_c\}$, the key space is 2^{2L}.
- **Initialization**:

  a) Iterate CCS for $\eta \geq \lceil \lambda_c \rceil$ times to obtain the pseudo-random perturbing interval Δ_c, which should be smaller than 2^n, and is a prime number.

  b) Iterate CCS for about 2^n times to obtain 2^n non-zero pseudo-random initial conditions $x_{e0}(1) \sim x_{e0}(2^n)$ for all ECS-es. The 2^n initial conditions are stored in CIT.

  Note: If zeros occur in pseudo orbit of CCS, then the number of iterations will be a little larger than 2^n. But the probability of such an event is small when $L \geq 16$ and $L \geq 2n$. For example, when $L = 16, n = 8$, the probability is about 0.00389866021632. There is another way to solve this trivial problem: once CCS goes to zero, use its perturbing PRNG to perturb it and then use the perturbed orbit to generate $x_{e0}(i)$. The latter way can make the iteration number always be 2^n.

  c) Iterate CCS for about 2^n times again to obtain 2^n non-zero (see above note) pseudo-random control parameters $p_{e0}(1) \sim p_{e0}(2^n)$ for all ECS-es. If there are at least two control parameters are same, discard all 2^n control parameters and re-initialize the ECS-es. The 2^n control parameters are also stored in CIT.

  Note 1: If L is not too larger than n, the probability of the occurrence of identical control parameters may be rather large to make the initializations slow. For example, when $n = 8, L = 16$, the probability is about 0.4. So it is desired that $L \gg n$, which can ensure the probability is near 0. For example, when $n = 8, L = 24$, the probability is only about 0.002.

  Note 2: Actually, the requirement on identical control parameters can be relaxed as follows: only when at least two ECS-es have the same control parameters and also the same initial conditions, initialization will be reset. Then the probability will be small enough so that we can neglect it in practice. For example, when $L = 16, n = 8$, the probability is about 0.00000765916767464514.

  d) Sort the 2^n initial conditions $x_{e0}(1) \sim x_{e0}(2^n)$ to generate the initial S-box (a pseudo-random permutation of $0 \sim 255$) $S[0] \sim S[2^n - 1]$. The sequence is used to initialize the perturbing intervals as follows:

$$\Delta_e(i) = \begin{cases} P_r(S[i]), & 0 \leq S[i] \leq I_{max} \\ P_r(\text{rand}(I_{max})), & I_{max} + 1 \leq S[i] \leq 2^n \end{cases}, \tag{9.3}$$

  where $P_r(0) = 1$ and $P_r(i)(i > 0)$ denotes the i^{th} prime number from 2 and $\text{rand}(n)$ means a random function to generate pseudo-random

integer between 1 and n. The I_{max} primes numbers $P_r(1) \sim P_r(I_{max})$ are pre-calculated and stored in CIT.

Note: A suggested value of I_{max} is 31 ($P_r(31) = 127$), which is enough to ensure long cycle length of the stream sub-cipher. See §9.4.2 for more details.

e) Iterate each ECS(i) for $\eta \geq \lceil \lambda(i) \rceil$ times, where $\lambda(i)$ is the Lyaponov exponent of ECS(i). Finally, iterate each ECS(i) one by one until NC_{max} L-bit pseudo-random integers are generated, which are then used to initialize $N_F(1) \sim N_F(NC_{max})$.

- **Encryption Procedure**: One *plain-cluster* is firstly encrypted by the stream sub-cipher, then by the block sub-cipher. We respectively depict how the two sub-ciphers work.
  - **Stream Sub-Cipher**: The stream sub-cipher encrypts the *plain-cluster* L-bit block by L-bit block. Assume the current L-bit plain-block is the i^{th} ($i = 1 \sim NC_{max}$) in the current plain-cluster $P_L(i)$. The encryption procedure can be denoted as follows: run CCS once, get $I_n = (x_c \bmod 2^n) + 1$, iterate ECS($I_n$) once, then output $\widetilde{P}_L(i) = P_L(i) \oplus x_e(I_n)^*$. Note that only the selected ECS is iterated once for the encryption of one plain-block, which is useful to promote the encryption speed and enhance the security of the stream sub-cipher. The encryption procedure goes until all plain-blocks in current *plain-cluster* exhaust, and then the *pre-masked plain-cluster* is sent to the block sub-cipher for encryption.
  - **Block Sub-Cipher**: The block sub-cipher is a simple substitution cipher with time-variant $n \times n$ S-box pseudo-randomly controlled by the 2^n ECS-es and the cluster buffer simultaneously. Divide L-bit variables $N_F(1) \sim N_F(NC_{max})$ into n-bit integers[†] $N_{Fn}(1) \sim N_{Fn}(NC_{max} \cdot L/n)$, and $\widetilde{P}_L(1) \sim \widetilde{P}_L(NC_{max})$ to $\widetilde{P}_{Ln}(1) \sim \widetilde{P}_{Ln}(NC_{max} \cdot L/n)$, then use the current S-box to substitute the pre-masked *plain-cluster* n-bit block by n-bit block as follows: $C_n(i) = S\left[\left(\widetilde{P}_{Ln}(i) + N_{Fn}(i)\right) \bmod 2^n\right]$ ($i = 1 \sim NC_{max} \cdot L/n$). After the current *plain-cluster* is encrypted, the S-box is re-generated by sorting the 2^n chaotic states $x_e(1) \sim x_e(2^n)$ ($S[0] \sim S[2^n - 1]$ store the rank result). Finally, copy $N_F(NC_{max}) \sim N_F(NC_{max} + 1 - N_p)$ to $N_F(0) \sim N_F(-(N_p - 1))$, and set $N_F(i+1) = N_F(i) \oplus \widetilde{P}_L \oplus \widetilde{P}_L(i - N_p)$ for $i = 0 \sim (NC_{max} - 1)$.

*If the last plain-block has $L' < L$ bits, just encrypt it with the highest L' bits of $x_e(I_n)$ and set left bits of $\widetilde{P}_L(i)$ to zeros. Of course, the stream itself should have mechanism to tell decoder the value of L'.

†The size of any plain-cluster should be divided exactly by n, otherwise some synchronization marks must be added and the cipher-video must have some specific format. When $n = 8$, it is rather easy to satisfy this requirement.

Note: Apparently, like in CBC mode, the first $NC_{max} + N_p$ L-bit plain-blocks are not so secure as following plain-blocks. So a randomly-generated $(NC_{max} + N_p) \cdot L$-bit IV should be added at the beginning of each plain-video (adding a pseudo-random frame is also OK).

After the encryption of the current *plain-cluster* is complete, the stream sub-cipher continues to encrypt the next *plain-cluster*. The encryption procedure goes until the plain-video exhausts.

- **Decryption Procedure**: Decryption is the inverse of the encryption (see Figure 9.4). The *cipher-cluster* is firstly decrypt by the block sub-cipher, where the S-box is the inversion of the one for encryption. Then the *pre-decrypted cipher-cluster* is decrypted by the stream sub-cipher.

From the above description, we can see CVES can also be considered as a simplified case of the extended model in Figure 9.3b: the second XOR is replaced by "mod 2^n" in f_S. Since the extra operation makes CVES securer*, it seems that triple-encryption in 9.3 may be really useful.

§9.3.3 Modified CVES Supporting Random Retrieval – RRS-CVES

In the above CVES, random retrieval of cipher-video cannot be supported, since the chaotic orbits of CCS and all ECS-es cannot be predicted only from the position of a *cipher-cluster* in the whole cipher-video. To decrypt a *cipher-cluster*, we must decrypt all *cipher-clusters* before it. That is to say, the original CVES can only support sequent retrieval, not random retrieval. Fortunately, we can make some modifications on the original CVES to add this function. The modified CVES is called Random-Retrieval-Supported CVES (RRS-CVES).

- **Initialization**: Besides the initialization operations a)∼e) in original CVES, the following three operations are added.

  a') Generating Reset Information: Run the CCS for $2 + 2^n$ times to generate two L-bit pseudo-random numbers p_+, x_+ and 2^n m-bit pseudo-random numbers $\tau_e(1) \sim \tau_e(2^n)$†, which are also stored in CIT. Here, $\tau_e(i)(i = 1 \sim 2^n)$ should satisfy the following requirements: $\gcd(\tau_e(i), 2) = 1$ and $\tau_e(i) \geq \tau_{min}$, where τ_{min} should not be very small. In regards to the selections of m and τ_{min}, we will give some details in §9.3.4. The $2 + 2^n$ extra pseudo-random numbers are used to reset the 2^n ECS-es frequently (which is useful to support random retrieval).

*The original CVES in [112] does not employ such an extra operation so that it is not secure enough.

†An m-bit number x' can be obtained from L-bit chaotic states x as follows: $x' = x \bmod 2^m$ or $x' = x >> (L - m)$.

b′) Generating Sequence of Chaotic Iterations: Sort the 2^n pre-defined control parameters $p_{e0}(1) \sim p_{e0}(2^n)$ to generate a rank sequence $r_e(1) \sim r_e(2^n)$, where $r_e(i) = 1 \sim 2^n$. The sequence is stored in CIT and will be used to control the chaotic iterations of the 2^n ECS-es.

c′) Initializing Iteration Counters: 2^n L-bit memory units $C_1(1) \sim C_1(2^n)$ are used to store the iteration numbers of the 2^n ECS-es. Another 2^n L-bit memory units $C_2(1) \sim C_2(2^n)$ are used to store the reset numbers of the 2^n ECS-es. Set the $2 \cdot 2^n$ L-bit memory units to zeros.

To sum up, for RRS-CVES, there are the following predefined data stored in CIT: 1) *Initial Conditions* – $x_{e0}(1) \sim x_{e0}(2^n)$; 2) *Control Parameters* – $p_{e0}(1) \sim p_{e0}(2^n)$; 3) *Perturbing Intervals* – $\Delta_e(1) \sim \Delta_e(2^n)$; 4) *Prime Numbers List* – $P_r(1) \sim P_r(I_{max})$; 5) *Reset Information* – $\tau_e(1) \sim \tau_e(2^n)$ and p_+, x_+; 6) *Rank Sequence for Chaotic Iterations* – $r_e(1) \sim r_e(2^n)$; 7) *Iteration/Reset Counters* – $C_1(1) \sim C_1(2^n)$ and $C_2(1) \sim C_2(2^n)$. In original CVES, only the first four ones are required.

- **Encryption Procedure**: In RRS-CVES, the stream sub-cipher is modified with reset mechanism, but the block sub-cipher is untouched at all. In RRS-CVES, $r_e(1) \sim r_e(2^n)$ is used to select an ECS to encrypt the current plain-block, instead of iterating CCS in original CVES: for the i^{th} plain-block, select ECS($r_e(i \bmod 2^n)$) as the current ECS. For any ECS(i), after it runs once, increase its iteration counter by 1: $C_1(i) + +$. If $C_1(i) \bmod \tau_e(i) = 0$, **reset** ECS($i$) as follows: $x_{e0}(i) = (x_{e0}(i) + x_+) \bmod 2^L$, $x_e(i) = x_{e0}(i)$, and $C_1(i) = 0, C_2(i) + +$. If $C_2(i) \bmod \tau_e(i) = 0$, **reset** ECS($i$) as follows: $p_{e0}(i) = (p_{e0}(i) + p_+) \bmod 2^L$, $p_e(i) = p_{e0}(i)$ and $C_1(i) = C_2(i) = 0$.
- **Decryption Procedure**: Make the same modifications like encryption procedure.

From the encryption procedure of RRS-CVES, we can see the following fact. Consider the cipher-video as a L-bit data-stream, if we know the position of one *cipher-cluster* in the L-bit stream, it is possible to reconstruct the corresponding states of all ECS-es and then decrypt the *cipher-cluster*, within considerable maximal time-out. Assume the position of the *cipher-cluster* is I_L, i.e., the total number of L-bit cipher-blocks before the *cipher-cluster* is I_L. We can reconstruct all 2^n ECS-es as follows:

1. $I_{ECS} = (I_L \bmod 2^n) + 1$, $I'_L = I_L / 2^n$;
2. $i = 1 \sim 2^n$: $I_{c1}(i) = I'_L / \tau_e(i)$, $I'_{c1}(i) = I'_L \bmod \tau_e(i)$, $I_{c2}(i) = I_{c1} / \tau_e(i)$;

3. $i = 1 \sim 2^n$: $x_e(i) = (x_{e0}(i) + I_{c1}(i) \cdot x_+) \bmod 2^L$, $p_e(i) = (p_{e0}(i) + I_{c2}(i) \cdot p_+) \bmod 2^L$;
4. $i = 1 \sim 2^n$: if $r_e(i) = 1 \sim I_{ECS}$, run ECS(i) for $I'_{c1}(i) + 1$ times; else (if $r_e(i) = I_{ECS} + 1 \sim 2^n$) run ECS($i$) for $I'_{c1}(i)$ times;
5. Decrypt the cipher-cluster as normal procedure.

We can see some pre-computation is used to reconstruct the current states of the 2^n ECS-es. Thus, the maximal time-out for random retrieval will be determined by the number of chaotic iterations in step 4:

$$\sum_{r_e(i)=1}^{I_{ECS}} (I'_{c1}(i) + 1) + \sum_{r_e(i)=I_{ECS}+1}^{2^n} I'_{c1}(i) = \sum_{i=1}^{2^n} I'_{c1}(i) + I_{ECS}. \tag{9.4}$$

Assume the consuming time for one chaotic iteration is τ_0, the maximal time-out τ will satisfy $2^n \cdot \tau_{min} \leq \tau/\tau_0 < 2^{n+m}$. In the §9.4.1, we will further discuss this problem.

§9.3.4 Configure CVES and RRS-CVES

To optimize CVES and RRS-CVES in practical applications, some parameters should be carefully configured.

At first, we should carefully select chaotic systems used in CVES. As I suggested again and again in this dissertation, PWLCM-s are still suggested in CVES to obtain the best performance.

The basic parameters are L, n and N_p. 1) L: Since the key space is 2^{2L}, L should be large enough to provide high security. In addition, to simplify the realization of CVES in digital computers, $L = 32$ or 64 is suggested. If higher key entropy is needed, extra secret key can be introduced. For example, $x_{e0}(1)$ and $p_{e0}(1)$ can be introduced as new secret parameters and use ECS(1) to generate $x_{e0}(2) \sim x_{e0}(2^n)$ and $p_{e0}(2) \sim p_{e0}(2^n)$; 2) n: Apparently, the realization complexity of CVES/RRS-CVES has positive exponential relation with n: $O(2^n \cdot L)$ bits memory is needed. Thus, n cannot be too large, and we suggest $n = 8$. 3) N_p: as we mentioned above, this parameter is used to ensure security against chosen plaintext attacks, $N_p \cdot L \geq 256$ is suggested.

It has been known that the perturbing parameters of the 2^n ECS-es and CCS are very useful to improve the degradation of digital chaotic maps. In [81], the authors stated that the perturbing intervals can be very large, such as 10^6 when $L = 40$. But we argued that they cannot be too large from strict cryptographic consideration*. When $I_{max} = 31$, the maximal perturbing interval is 127 (31^{st}

*Consider the following fact: even when L is large enough, there always exist some chaotic orbits leading to short cycle length. An extreme example is the digital tent map $F(x) = 1 - 2|x - 0.5|$, any orbits from $a/2L$ will lead to zero after at most L iterations.

prime number not less than 2), which is acceptable in practice.

The size of one *cluster* is another important parameter of CVES/RRS-CVES. Although the size needn't be fixed, the fixed-size *cluster* is useful to simplify the realization and the performance estimation. Assume a *cluster* contains NC_{max} L-bit blocks. We will conclude that NC_{max} can be used to adjust the encryption speed. Generally speaking, the larger NC_{max} is, the faster the encryption speed will be. Further details will be given in §9.4.1. If the *cluster* size is variant, the average size $\overline{P}_{max}$ can be used to estimate the encryption speed.

For RRS-CVES, m and τ_{min} are used to control the reset operations of the 2^n ECS-es. Generally, we suggest $m \leq n$ and $\tau_{min} \geq 2^{n/2}$. Then the maximal time-out will satisfy $2^{3n/2} \leq \tau/\tau_0 \leq 2^{2n}$. Since n is not too large, such a maximal time-out can be acceptable in most real-time applications (see §9.4.1 for more details).

§9.4 Performance Estimation

§9.4.1 Speed

We can estimate the encryption speed based on the speed of the two sub-ciphers. Generally speaking, the hardware system of CVES/RRS-CVES will run much faster than the software system, considering the parallel mechanism can be used in hardware realization. Without loss of generality, assume all ECS-es and CCS are realized by Eq. (2.1), and the *cluster* size is fixed: $NC_{max} \cdot L$ bits.

Hardware realization: Generally speaking, one L-bit fixed-point division consumes L clock cycles, then one chaotic iteration approximately consumes L clock cycles. Consider the multiple pipelining techniques can be used here, the stream sub-cipher encrypts one L-bit plain-block per L clock cycles. Assume the time consuming by the sorter is τ_s (clock cycles), for the most time-consuming sorter, $\tau_s = 2^n \cdot (2^n - 1)$; and for the optimized sorter using the quick sorting algorithm, $\tau_s = n \cdot 2^n$. In addition, the block sub-cipher encrypts one n-bit plain-block per clock cycles. To sum up, for one *plain-cluster*, the total consuming clock cycles is $L \cdot NC_{max} + \tau_s + NC_{max} \cdot L/n$, where τ_s denotes the time consuming by the sorter. If the basic clock frequency is f_b MHz, the final speed of CVES will be $f_b \Big/ \left(1 + \frac{1}{n} + \frac{\tau_s}{L \cdot NC_{max}}\right)$ Mbps. Apparently, NC_{max} can adjust the encryption speed. When $L = 32, n = 8$, $\tau_s = n \cdot 2^n$ and $NC_{max} = n \cdot 2^n = 2048$, the encryption speed is $\frac{32}{37} f_b$ Mbps. Such a speed is faster than many fast conventional ciphers. Of course, the estimated speed here is just a theoretical value, and the actual speed will be tightly depend on details of implementation.

From the above discussion, we can see the actual encryption speed is chiefly determined by the stream sub-cipher when $NC_{max} \geq \tau_s/L$, and by the sorter

in the block sub-cipher if $NC_{max} < \tau_s/L$. For most applications of CVES/RRS-CVES, we suggest $NC_{max} = n \cdot 2^n$. When $L \geq 32, n = 8$, NC_{max} will be larger than τ_s/L (even for the most time-consuming sorter, $\tau_s/L = 2^n \cdot (2^n - 1)/L < n \cdot 2^n$). Hence, in most conditions, the sorter can be realized chiefly from the consideration of simplifying the hardware scale, not promoting the sorting speed.

If extra cluster buffers are used to support pipeline encryption/decryption, then the encryption speed will be only determined by the slower sub-cipher, i.e., the speed will be $\min\left(f_b, f_b \Big/ \left(\frac{1}{n} + \frac{\tau_s}{L \cdot NC_{max}}\right)\right)$ Mbps. When $NC_{max} \geq \tau_s/L$, the speed will be about f_b Mbps.

Software realization: Software realization will be much slower than hardware realization since generally parallel mechanism is not available for software realization. It can be approximately evaluated that the speed of software realization will be several times slower than the hardware realization. An experimental system with parameters $L = 16, n = 8$ is designed with Microsoft® Visual C++ to test the actual speed under Microsoft® Windows™ platform. The final speed is about $1/L$ of the CPU frequency. For example, on a 1.4GHz Pentium®IV CPU, the speed is about 83Mbps; on a 700MHz Celeron® CPU, the speed is about 46Mbps. Such a speed is rather high for a software cipher (especially for a chaotic cipher). In comparison, the reference speed of 256-bit AES on a 600 MHz CPU is about 66 Mbps[166].

In our experimental system, we find that the stream sub-cipher, whose kernel is the digital PWLCM-s, plays leading factor on the final speed. On the above 1.4GHz Pentium®IV CPU, the speed of the stream sub-cipher is about 107Mbps, but the speed of the block sub-cipher is about 636Mbps! Comparing the speed of the stream sub-cipher (107Mbps) to the final speed (83Mbps), it seems that our C++ codes can be further optimized to realize faster speed (near to 100Mbps). A strong cue is found when we change the compiling switch from "default" to "Maximized speed" in Visual C++ 6.0, the speed of the stream sub-cipher become 98Mbps. This strange fact that "maximized speed" is lower than "default" speed implies our C++ codes should be specially optimized. In future I will try to find more results on this issue.

Finally, let us discuss the time consuming on the initialization and the time-out problem of RRS-CVES.

For the initialization of CVES, the most time-consuming procedure is about $(2 + \eta) \cdot 2^n$ chaotic iterations and a sorting procedure of 2^n data, which means $(2 + \eta) \cdot 2^n \cdot L + n \cdot 2^n$ clock cycles. Assume $\eta = 4 > \lceil \lambda \rceil = 2$, the consuming time will be about $(6L + n) \cdot 2^n$ clock cycles. When $L = 32, n = 8$, the time is $51,200$ clock cycles. Similarly, for RRS-CVES, the consuming time of chief procedure can be calculated to be about $(3 + \eta) \cdot 2^n \cdot L + 2 \cdot n \cdot 2^n = ((3 + \eta) \cdot L + 2n) \cdot 2^n$ clock

cycles. Assume $\eta = 4$ and $L = 32, n = 8$, the time will be 61,400 clock cycles. Obviously, the initialization will not consume too much time.

For the maximal time-out for random retrieval of RRS-CVES, we have pointed out that $2^{3n/2} \leq \tau/\tau_0 \leq 2^{2n}$ in §9.3.4. When $L = 32, n = 8$, we can get $2^{17} \leq \tau \leq 2^{21}$ clock cycles (consider τ_0 equals to L clock cycles). If $f_b \geq 200$ MHz, the maximal time-out will be not greater than 10ms.

§9.4.2 Security

Basic Cryptographic Properties

CVES has the following basic cryptographic properties, which are basic requirements for good ciphers.

1) Balance: Since the chaotic orbits of the 2^n ECS-es have uniform distribution function, then the *plain-clusters* pre-masked by the stream sub-cipher will also have uniform distribution function. Consider the block sub-cipher subsequently substitutes the *pre-masked plain-clusters*, which cannot change the uniform distribution because the substitution operation with S-box is a bijective map. Consequently, the cipher-video will be balanced.

2) Avalanche Property with Respect to Secret Key: If the secret key $K = \{x_c, p_c\}$ changes only one bit, then the initial conditions or control parameters of ECS-es will change much because CCS' sensitivity to initial conditions and control parameters. The initial conditions and/or control parameters change a little, the ciphertext will change much, which implies the avalanche property of ciphertext.

Essentially Features to Avoid Potential Attacks

There are four essential features to imply CVES' security. Of course, more further studies will be made to find new evidences.

1. The chosen plaintext attack mentioned above has been disabled by feedback of N_p L-bit plain-blocks, which makes the time-variant S-box f_S dependent on both the chaotic states and the previous plaintexts.

2. The statistical cryptanalytic tools to catch defects of weak S-box are useless, since almost every plain-cluster has a different S-box and the cluster size is too small (generally $O(2^8)$) to expose such defects.

3. The stream sub-cipher is made of 2^n asymptotically independent chaotic maps (ECS Pool), and the sequence of chaotic iterations is controlled

pseudo-randomly by another independent chaotic map (CCS). The above two facts make the statistical cryptanalysis much more difficult.

4. Extremely, even when the 2^n states of ECS pool are all known (then the related S-box is also known)*, it is impossible to derive the secret key $K = \{x_c, p_c\}$ since the key is separated from the 2^n states by previous chaotic iterations of 2^n ECS-es. Please recall the initialization procedure, $\eta \geq \lceil \lambda \rceil$ pre-iterations are required, which is used to avoid the above (potential) attack if the 2^n chaotic states of the first *cluster* are known.

Cycle Length of the Stream Sub-Cipher

In CVES/RRS-CVES, both the stream sub-cipher and the block sub-cipher are based on the digital orbits of the 2^n ECS-es and CCS. Consider the current 2^n states of ECS pool as a 2^n-dimensional vector (here we call it Chaotic Vector) , the cycle length of this vector will be a crucial factor to measure the security of the whole system. The cycle length should be large enough to avoid repeated encryption pattern.

From [81], we can easily get the cycle length of CCS: $T_c = \sigma_c \cdot \Delta_c \cdot (2^L - 1)$, and the cycle length of 2^n ECS-es: $T_e(i) = \sigma_e(i) \cdot \Delta_e(i) \cdot (2^L - 1)(i = 1 \sim 2^n)$, where $\{\sigma_e(i)\}_{i=1}^{2^n}, \sigma_c$ are positive integers. Although it is difficult to measure the exact cycle length of Chaotic Vector, we can derive its order:

$$\begin{aligned} \operatorname{lcm}\left(\sigma_e(1), \cdots, \sigma_e(2^n), \sigma_c\right) \cdot \left(2^{L_1} - 1\right) \cdot \left(2^{L_2} - 1\right) \cdot \operatorname{lcm}\left(\Delta_e(1), \cdots, \Delta_e(2^n), \Delta_c\right). \\ > 2^{L_1 + L_2} \cdot \operatorname{lcm}\left(\Delta_e(1), \cdots, \Delta_e(2^n), 2\right). \end{aligned} \tag{9.5}$$

When $I_{max} = 31$,

$$\operatorname{lcm}\left(\Delta_e(1), \cdots, \Delta_e(2^n), 2\right) = \prod_{i=1}^{I_{max}} P_r(i) \approx 2^{161}. \tag{9.6}$$

Such a length is large enough for any secure applications.

Pseudo-Random S-Boxes of the Block Sub-Cipher

In this subsection, we discuss the statistical properties of S-boxes pseudo-randomly generated by the chaotic states of ECS pool. Because the 2^n ECS-es have the same invariant density function $f(x) = 1$ on the same interval $I = [0, 1]$, the generated S-boxes by sorting the 2^n chaotic states can be depicted as the *rank*

*Actually, in practice, such an attack is rather difficult since the current chaotic states cannot be obtained from a pair of *plain-cluster* and *cipher-cluster*.

statistics of 2^n random variables with identical and independent distribution functions. Let $R(1), R(2), \cdots, R(2^n)$ denote the rank statistics, then the following fact is true: for any permutation $\{i(1), i(2), \cdots, i(2^n)\}$ on $\{1, \cdots, 2^n\}$,

$$P\{R(1) = i(1) \wedge R(2) = i(2) \wedge \cdots \wedge R(2^n) = i(2^n)\} = \frac{1}{2^n!}, \tag{9.7}$$

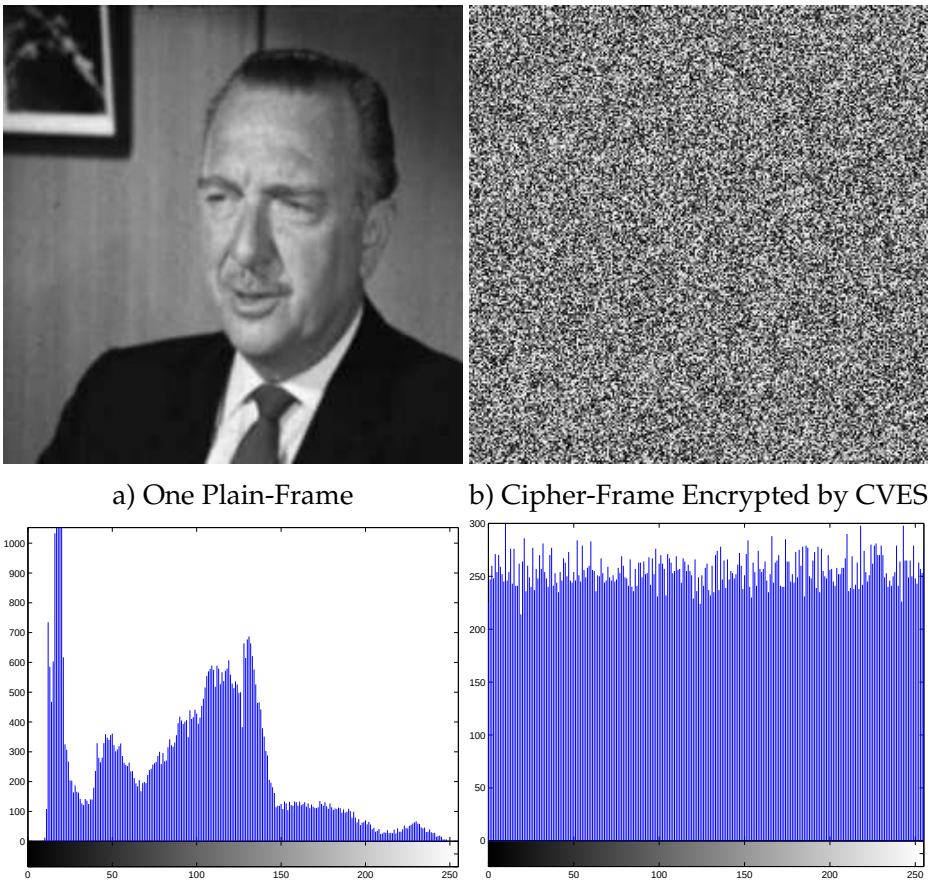
i.e., the rank statistics is equiprobable and symmetric[212]. It is very useful to construct ciphers with perfect cryptographic properties. Of course, there really exist many weak S-boxes in all $2^n!$ possible ones, but the number will be much smaller than the strong ones. What's more, the product of stream sub-cipher and block sub-cipher makes the detection of weak S-boxes difficult. Under the worse condition, if one weak S-box is broken, only the related *plain-cluster* will be influenced, all other *plain-clusters* with different S-boxes will still keep secure.

§9.4.3 Realization Complexity

Because generally L and n can be divided exactly by 8, the software realization of CVES/RRS-CVES will be very simple, since 8-bit byte is supported well by almost all programming languages under different platforms. Therefore, we focus on the realization complexity by hardware in this subsection.

The most important hardware devices are one L-bit digital dividers to iterate the digital chaotic systems and a $2^n \times 2^n$ sorter. Other devices include: two m-LFSR-s, and some memory units to store the CIT, current 2^n chaotic states, cluster buffer and the S-box. For CVES, the CIT needs $4 \cdot 2^n$ L-bit memory units and the S-box needs 2^n n-bit memory units. For RRS-CVES, the CIT needs $8 \cdot 2^n$ L-bit memory units and the S-box still needs 2^n n-bit memory units. When $L = 64$, $m = 8$, the total number of memory units of CVES is about $4 \cdot 2^n \cdot L + n \cdot 2^n = 67,584$ bits $= 8,448$ bytes. For RRS-CVES, the total number is $9 \cdot 2^n \cdot L + n \cdot 2^n = 149,504$ bits $= 18,688$ bytes. In addition, a data buffer whose size equals to the *cluster* size may be also needed to facilitate the substitution of the block sub-cipher after the encryption made by the stream sub-cipher. Although the required memory is relatively large (about tens KB), but for real-time video applications it should be not worth mentioning at all.

In CVES/RRS-CVES, the sorter is the most complicated device. As we have mentioned in §9.4.1, when the cluster size is generally larger than $n \cdot 2^n / L$, the sorter can be realized without too many considerations, since the sorter makes only a little influence on the encryption speed.



a) One Plain-Frame b) Cipher-Frame Encrypted by CVES

c) Histogram of the Plain-Frame d) Histogram of the Cipher-Frame

Figure 9.5: Uncompressed Digital Video Encrypted with CVES

§9.4.4 Experiments

For an uncompressed digital video, we test the practical performance of CVES. In Figure 9.5, we give the comparison of one plain-frame and the cipher-frame. We can see the plain-image is encrypted to a cipher-image with uniform histogram, which implies the perfect cryptographic properties of CVES.

§9.5 Conclusion

In this chapter, we propose a new encryption scheme (Chaotic Video Encryption Scheme – CVES) for real-time digital video based on multiple digital chaotic maps, which is a trial of solving the paradox between the encryption speed and high security of real-time video encryption. CVES is a product cipher that con-

tains a stream sub-cipher and a block sub-cipher. CVES can be extended to RRS-CVES, an enhanced version supporting random retrieval of cipher-video with considerable maximal time-out. Initial analyses have shown that CVES/RRS-CVES has fast speed and acceptable security, and can be realized easily by both hardware and software. Actually, the basic idea used in CVES can be generalized as a common model of digital ciphers shown in Figure 9.1.

In the future, we will investigate the further issues about the security and realization of CVES/RRS-CVES and try to complete the standard realization packages with VLSI (hardware) and C/C++ language (software), and study how to use ideas in other video encryption methods to enhance the performance of CVES in specific video applications. If possible, the two extended models in Figure 9.3 will be focused.

Chapter 10
Conclusion and Remarks on Future Research

§10.1 A Summary of this dissertation

Now let us give a summary on this dissertation. Our works described in this dissertation can be classified into three parts: theory on digital chaos, cryptanalyses of digital chaotic ciphers, and new paradigms to design digital chaotic ciphers. In the following we would like to separately summary our works in the three directions.

- **Theoretical Analyses of Dynamical Degradation of Digital Chaotic Systems**
  - It is very important to develop some theory to rigorously depict dynamics of digital chaotic systems. Although many works are made to explore dynamical properties of digital chaotic systems, no systematic methods are constructed. For a class of widely-used chaotic systems - piecewise linear chaotic maps (PWLCM-s), a series of dynamical indicators are found to quantitatively measure the non-uniformity of distribution of pseudo orbits. Exact calculation of the dynamical indicators are theoretical deduced and some examples are given to show their significance in understanding dynamical degradation of digital PWLCM-s in finite precision. Applications of the proposed dynamical indicators are discussed in detail, including their use in chaotic cryptography (see below).
- **Cryptanalyses of a Number of Recently-Proposed Digital Chaotic Ciphers**
  - Using the proposed dynamical indicators of PWLCM-s, some digital chaotic stream ciphers proposed by Hong Zhou et al. are cryptanalyzed with weak-key analyses. A comparison of possible solutions to enhance attacked ciphers is given and several ones are suggested to enhance security of attacked ciphers.
  - In 1999, E. Alvarez et al. proposed a chaotic cipher based on searching plaintext in a pseudo-random sequence generated from chaotic systems, but soon it was broken by G. Alvarez et al. in 2000. This dissertation analyzes why E. Alvarez et al.'s cipher is so vulnerable to G. Alvarez et al.'s attacks, and proposes a modified scheme to enhance the security of the original cipher.

  - In 1998, M. S. Baptista proposed a searching based chaotic cipher, which attracted much attention after its proposal. Some cryptanalytic works and modifications are made in recent years. This dissertation points out the deficiency of an attack proposed by Goce Jakimoski and Ljupčo Kocarev, and presents a remedy to resist all known attacks (not only Jakimoski-Kocarev attack). In the proposed remedy, an interesting feature called probabilistic decryption is found, which may be helpful to realize another type of visual cryptography.
  - In 2001, S. Papadimitriou et al. proposed a probabilistic cipher based on chaotic systems with fast speed. This dissertation analyzes problems of this chaotic cipher and points out its insecurity and impracticalness. Some wrong deduction and analyese given by S. Papadimitriou et al. are also be rectified.
  - In recent years, J.-C. Yen and J.-I. Guo et al. proposed several chaotic image encryption methods. This dissertation cryptanalyzes two Yen-Guo chaotic image encryption methods (CKBA and BRIE), and proposes known/chosen plaintext attack to break the two systems.

- **New Proposals to Design Digital Chaotic Ciphers**
  - Based on theoretical results on digital chaotic systems and cryptanalyses of several recently-proposed chaotic ciphers, this dissertation proposes a new chaotic PRBG and uses it to design chaotic stream ciphers with better overall performances. The proposed chaotic PRBG can be used instead of LFSR in conventional stream-cipher cryptography to construct more flexible ciphers.
  - Still based on the above achievements, this dissertation proposes a chaotic cipher with very fast encryption speed and can be used to fulfill needs of real-time video encryption. Detailed analyses show that the proposed chaotic cipher can provide rather fast encryption speed and high level of security simultaneously. The cipher can also be considered as a general model of new digital (chaotic) ciphers.

§10.2 Perspective of Future Research

From discussions given in this dissertation, we can find the design of really good digital chaotic ciphers is not a easy task. Many issues must be considered carefully to avoid potential security defects and to get desired performance. However, since theoretical issues about dynamical degradation of digital chaotic systems has not been solved, we can only use some practical remedies carefully to

circumvent this theoretical difficulty. As a suggestion, the pseudo-random perturbation algorithm seems to be an acceptable remedy. Besides security problems about dynamical degradation, many digital chaotic ciphers are broken because of their careless design, not because of the essential defects of digital chaotic systems. This fact shows some design principles about how to avoid designing a weak chaotic cipher should be developed. Here we will give some common suggestions on digital chaotic cipher. Hope my suggestions can be helpful to expedite systematization of such design principles.

§10.2.1 Suggestions for the Design of a Good Chaotic Cipher

Based on the review to the state-of-the-art of digital chaotic ciphers, some problems and possible solutions, at the end of this dissertation I would like to re-argue some fundamental suggestions for the design of a "good" chaotic cipher, where the term "good" means high practical security, fast encryption speed and simple realization.

Suggestion 1 – Realizing digital chaotic systems via pseudo-random perturbation, or using discretized chaotic systems whose dynamical properties have been proven. As we have mentioned in §2.5, there exists degradation on the dynamical properties of digital chaotic systems realized in finite precision. Under the situation that no systematic theory to measure such degradation, some remedies must be adopted to improve the dynamical properties of digital chaos. The perturbation algorithm by a simple PRNG is suggested by me, since it has considerable practical performance. The discretized versions of some continuous-value chaotic maps may be also OK, but it is desired that the designers prove (at least "explain with some experimental evidences") their dynamical (cryptographic) properties.

Suggestion 2 – Using fixed-point arithmetic instead of floating-point arithmetic. It is obvious that floating-point arithmetic will lower the encryption speed and increase the realization complexity and cost. Thus, fixed-point arithmetic is suggested. In addition, the fixed-point arithmetic is also helpful to improve the portability between different software platforms or hardware structures. There are another defect about floating-point arithmetic: the discretized lattice of floating-point arithmetic is not uniformly, which will make it much more complicated and difficult to control the degradation of digital dynamical properties.

Suggestion 3 – Using the simplest chaotic systems, such as piecewise linear chaotic maps (PWLCM-s). More complicated chaotic systems are usually suggested being used to ensure the security of developed chaotic ciphers. But

the use of complicated chaotic systems will lower the encryption speed twofold: i) the more complicated the chaotic maps, the more time the chaotic iterations will consume; ii) many complicated chaotic systems must run with floating-point arithmetic, which makes the iterations further slower. Basically, we suggest using PWLCM-s in all chaotic ciphers. If PWLCM-s cannot be used in some applications (we think such applications are rare), choose the simplest chaotic systems that are available.

Suggestion 4 – Avoiding the use of multiple iterations for one ciphertext. The slow encryption speed of most chaotic block cipher is chiefly determined by the use of multiple iterations for one ciphertext. Most known chaotic block ciphers do not yield this suggestion. Several new chaotic block ciphers[105, 106, 108, 112, 124] overcome this problem and can be used for reference.

Suggestion 5 – Using multiple chaotic systems instead of one single one. Although no rigorous proof is given, knowledge (lessons and experience) from the design of good and bad chaotic ciphers implies the use of multiple chaotic systems to enhance its security. Some studies[22, 112, 124] have shown that the use of multiple chaotic systems can also promote encryption speed.

§10.2.2 Open Topics in Cryptography based on Digital Chaos

In [101], L. Kocarev suggested that the future research in chaotic cryptography should focus on the relationships between chaos and cryptography, not the *ad hoc* design of new chaotic ciphers. Basically, we agree to his opinion. Of course, new structures of chaotic ciphers may still be useful, if some really novel ideas are introduced and much better performance is provided. The following are some open topics in chaotic cryptography.

Theory about digital chaos. To estimate the dynamical properties of digital chaotic systems, a systematic theory about chaos in discrete space is needed. However, there are only a few efforts made in this direction. In [202], H. Waelbroeck et al. tried to translate the definitions of continuous chaos to the context of discrete state space (called "discrete chaos"). It is an interesting works on this way. Our analyses given in Chap. 3 and [109] also show a new way to study digital chaos from an arithmetic point of view.

Unpredictability of the pseudo-randomness generated by digital chaos. The pseudo-random sequences generated by digital chaos are kernel parts in many chaotic ciphers. How to measure the unpredictability of the pseudo-random sequences is an unsolved problem. In continuous chaos theory, *information entropy* can be used to depict the rate of the information loss as the chaotic iterations go[206, 209, 242]. Similar concept may be also used to qualitatively explain

the unpredictability, such as the analyses in [22, 61].

Chaos in conventional ciphers. We have mentioned any conventional cipher can be considered as a chaotic or pseudo-chaotic cipher in §1.1. Some chaotic behaviors hiding in conventional ciphers have been reported by W. Schwarz et al.[21]. In the future research, the following investigations will be useful for the design of conventional ciphers and chaotic ciphers: 1) Can we use chaos theory to explain the nonlinear functions and operations used in conventional ciphers? For example, can the mod function defined on finite filed be considered as a discretized chaotic map*? 2) Can we re-define the confusion and diffusion with chaos theory? Can we find a way to connect the security measurement (such as linear complexity in stream-cipher cryptography) in conventional cryptography with the measurements (such as the information entropy) in chaos theory?

General models for the design of digital chaotic ciphers. Since several general models have been proposed, further efforts on the proposed models will be helpful to exploit the relationship between chaos and cryptography. Of course, new general models are also wanted.

Cryptanalyses of known digital chaotic ciphers. As we know, the recent advances in block-cipher cryptology are promoted by the emergence of the differential and linear cryptanalysis, which shows the importance of the cryptanalysis in cryptology[144, 145]. We believe any new attacks of some chaotic ciphers will impulse the progress of chaotic cryptography.

*Consider the digital tent map realized in fixed-point discrete space.

Acknowledgements (致谢)

All achievements contained in this thesis should be firstly owed to my supervisors: Prof. Yuanlong Cai and Prof. Xuanqin Mou. It is their serious supervision and encouragement to help me overcome many difficulties during the 4-year life towards the final completion of this thesis. When I was still a master student in the Institute of Image Processing of Xi'an Jiaotong University, I had been already directed by Prof. Cai and Prof. Mou. I benefit a lot from their rich knowledge and research experience, and such a great influence will follow me in all my life. The following facts deserve mentioned for special acknowledgements: without their warmly encouragements and continuous support, it was impossible for me to change my research direction from medical imaging (and later intelligent transportation systems - ITS) to the one presented in this thesis, chaotic cryptography, in 2000. In addition, during the six years of my study in the Institute of Image Processing, Prof. Mou gave me a lot of help in many aspects of my everyday life besides the research itself. I would like to say, Prof. Mou is not only a good teacher, but also a good friend.

Special thanks are given to Prof. Yumin Wang with the Xidian University, who gave me a lot of help on my initial research and make me towards a right way pursuing the research. Thank Prof. Jianxue Xu and Dr. Xiangao Huang with the Xi'an Jiaotong University, for helpful discussions with them. Thank Prof. Zhengjin Feng with the Shanghai Jiaotong University and some of his PhD students – Dr. Lihui Zhou, Dr. Guojie Hu and Mr. Guangliang Cui, for the private communications with them that helped the author understand their research work better.

Thank Prof. Guanrong Chen with the City University of Hong Kong, for frequently sending latest references to me. Also, thank him for inviting me to do further visiting research on chaos-based image/video encryption with him after my graduation. Thank Dr. K.-W. Wong with the City University of Hong Kong, who helped me know how the cipher proposed in [123] works exactly. Thank Dr. Cunsheng Ding with the Hong Kong University of Science and Technology for his help on my paper [22].

Thank Dr. Goce Jakimoski with the University of California, San Diego (UCSD), whose comments on my article [128] drove me to find out the security defect of the originally-proposed bit-extracting functions and to figure out new countermeasures. Thank Prof. Ljupčo Kocarev with the UCSD, whose e-mails motivated me to double check the analyses given in [128], leading to the finding of the so-called "probabilistic decryption" effect existing in [128]. Thank Dr.

Gonzalo Álvarez with the Instituto de Física Aplicada, Consejo Superior de Investigaciones Científicas, Spain, who sent his unpublished paper [126] to me and gave many encouragements on my research.

Thank Dr. S. Papadimitriou with the University of Patras (Greece) for his sending the C++ codes of the chaotic cryptosystem proposed in [106] to the author. Thank Prof. Richard A. Stong at the Rice University and Dr. Jeb Faulkner Willenbring with the Yale University for their help on the proof given in the Appendix of Chap. 6.

Thank an anonymous reviewer for his/her valuable comments on my ignorance on previous related work in a submission to *Int. J. Bifurcation and Chaos*. He/She pointed out several important references that I did not notice at that time (especially Prof. M. Blank's monograph [172]), which helped me find a lot of new references on dynamical properties of digital chaotic systems and make the content of §2.5.1 much more comprehensive and accurate.

Sincere thanks are also given to the following researchers and friends for their help on finding some references: Prof. P. E. Kloeden with the Johann Wolfgang Goethe University (Germany), Prof. Jiri Fridrich with the Thomas J. Watson School of Engineering and Applied Science, Prof. Josef Scharinger with the Johannes Kepler Universität (JKU) Linz (Austria), Prof. Reihaneh Safavi-Naini with the University of Wollongong (Australia) and her students Mr. Chandrapal Kailasanathan, Mr. Takeyuki Uehara, Dr. Ching-Nung Yang with the National Dong Hwa University (Taiwan, China), Prof. Der-Chyuan Lou with the National Defense University (Taiwan, China), Ms. Yifang Xu with the Duke University, Ms. Wei Feng with the National University of Singapore, Ms. Zhi Li with the Hong Kong University, Mr. Zhijun Zhang with the Chinese University of Hong Kong, Mr. Jinhua Li with the Borland Inc. of Singapore, Mr. Huaifeng Zhang with the Chinese Academy of Sciences. Their help made me much easier know latest progress in chaotic cryptography, image and video encryption, and some other related research areas. Thank Mr. Yalong Hu with the Chinese Academy of Sciences, who offered me very convenient accommodation and Internet access service during the time I was spending in Beijing in Jan. 2000 for collecting references.

Thank Dr. Zhen Ji, Prof. Jihong Zhang with the Shenzhen University, Mr. Boliya L. Yang with the Nankai University, Mr. Xuan Zheng with the Univeristy of virginia, Mr. Qi Li with the University of Liverpool (UK), and Ms. Wenmin Li with the Imperial College (UK), for their cooperations on related research. Thank Mr. Qi Li and Mr. Xuan Zheng for making presentations for me in the conferences IMA C&C'2001 and EI'2002, respectively.

Thank my roommates in the Xi'an Jiaotong University – Dr. Pinyi Ren and

Mr. Huayao Zhang – for their help in life and their comments on my research. Thank Dr. Yu Zhang with the Fudan University (xy@bmy) for his valuable words on how to do good research and for sending me an electronic copy of the *Zhong-Tu Classification System* (中图分类号). Thank Dr. Yongjin Zhou (BPMF@bmy) and Dr. Dong Dai (kenny@bmy) with the Xi'an Jiaotong University, whose discussions with me were very helpful to promote my research. Thank Mr. Shuangliang Di, Mr. Xin Fan and Mr. Hua Huang with the Institute of Image Processing, Xi'an Jiaotong University, who also gave me much help.

Thank Mr. Xiaolin Wang with the Air Force University of Technology (空军工程大学), Mr. Xiang Zhu with the Xi'an Jiaotong University, who gave a lot of help on my work and my life at the Institute of Image Processing, Xi'an Jiaotong University. Also, thank other teachers with the Institute of Image Processing: Mr. Chunhua Du, Dr. Chun Qi and Mr. Yaojin Zhao. I got many encouragements and helps from all of them during the six-year study life. Finally, I would like to thank all guys studying in the same research group. The life became quite interesting and bright because of the friendship with them.

Thank Dr. Harry Shum with the Microsoft Research Asia (MSRA), who invited me to do five-month research as a visiting student in 2002. The experience in MSRA greatly promote my capability of research. Thank Prof. Manuel Blum with CMU (Carnegie Mellon University), whose suggestions and opinions on SecHCI were very valuable for me to go further. Thank the following friends for my happy life in MSRA: Dr. Jian Sun, Dr. Tianshu Wang, Dr. Hong Chen, Ms. Lin Liang, Mr. Ping Tan, Mr. Lei Wang, Ms. Bei Lei, Mr. Wei Feng, Ms. Liang Wan, Mr. Bo Sun, Mr. Ziqiang Liu, Mr. Dongsheng Wang, Mr. Yunshu Hou, Ms. Yuanzhen Li, Mr. Jinyu Li, Mr. Yihua Xu, Mr. Ming Su and Mr. Yue Lu.

Thank Mr. Shuangliang Di with the Xi'an Jiaotong University because he led me into the wonderful world of TEX. Thank all CTEXers in ChinaTEX and CTEX, since it was impossible for me to complete this thesis successfully without their help. Thank Dr. Tianshu Wang with the IBM R&D Center (China) for his composing and publishing a TEX template of Ph. D. thesis of Xi'an Jiaotong University, which made me write the TEX document class xjtuthesis much more easily. Thank Ms. Yanghui Cao with the Xi'an Jiaotong University and Ms. Lu Han with the Xi'an Institute of Foreign Languages, who helped me enhance the grammar of many papers.

At last, I would like to give the most sincere thanks to my parents, whose teaching and education gave me great courage and confidence never to give up. And I also would like to thank my dearest girlfriend, Ms. Songqin Xie, without whose continuous support and considerate care no any work (including this thesis itself) can be completed successfully.

Bibliography (参考文献)

[1] Jules Henri Poincaré. *New Methods of Celestial Mechanics: Part 3. Integral Invariants and Asymptotic Properties of Certain Solutions*. History of Modern Physics and Astronomy, vol. 13. Springer Verlag, 1992.

[2] A. N. Sharkovskii. Coexistence of cycles of a continuous map of a line into itself (in Russian, English summaries). *Ukrainskii Matemacheskii Zhurnal (Ukrainian Mathematical Journal)*, 16(1):61–71, 1964.

[3] Tien Yien Li and James A. Yorke. Period three implies chaos. *American Mathematical Monthly*, 82(10):985–992, 1975.

[4] David Ruelle. Strange attractor. *The Mathematical Intelligencer*, 2(1):126–137, 1980.

[5] A. N. Sharkovskii. Coexistence of cycles of a continuous map of a line into itself. *Int. J. Bifurcation and Chaos*, 5:1263–1273, 1995.

[6] Otton E. Rössler. An equation for continuous chaos. *Physics Letters A*, 57(5):397–398, 1976.

[7] M. Hénon. A two dimensional mapping with a strange attractor. *Communications in Mathematical Physics*, 261:459–467, 1976.

[8] Mitchell J. Feigenbaum. Quantitative universality for a class of nonlinear transformations. *J. Statistical Physics*, 19(1):25–52, 1978.

[9] Robert M. May. Simple mathematical models with very complicated dynamics. *Nature*, 261:459–467, 1976.

[10] A. M. Zhabotinsky. Periodic liquid phase reactions. *Proc. Ac. Sci. USSR*, 157:392–395, 1964.

[11] A. N. Zaikin and A. M. Zhabotinsky. Concentration wave propagation in two-dimensional liquid-phase self-oscillating system. *Nature*, 225:535–537, 1970.

[12] Edward N. Lorenz. Deterministic non-periodic flow. *J. Atmospheric Sciences*, 20:130–141, 1963.

[13] Edward N. Lorenz. The predictability of hydrodynamic flow. *Trans. NY. Academy of Sciences Series II*, 25:409–432, 1963.

[14] James Gleick. *Chaos: Making a New Science*. Viking Penguin, New York, 1987.

[15] Ian Stewart. *Does God Play Dice?: The Mathematics of Chaos*. Blackwell Publishers, Oxford, UK, 1990.

[16] Edward N. Lorenz. *The Essence of Chaos*. University of Washington Press, 1993.

[17] R. Brown and L. O. Chua. Clarifying chaos: Examples and counterexamples. *Int. J. Bifurcation and Chaos*, 6(2):219–249, 1996.

[18] Jiri Fridrich. Symmetric ciphers based on two-dimensional chaotic maps. *Int. J. Bifurcation and Chaos*, 8(6):1259–1284, 1998.

[19] Ljupčo Kocarev, Goce Jakimoski, Toni Stojanovski, and Ulrich Parlitz. From chaotic maps to encryption schemes. In *Proc. IEEE Int. Symposium Circuits and Systems 98*, volume 4, pages 514–517. IEEE, 1998.

[20] G. Alvarez, F. Monotoya, G. Pastor, and M. Romera. Chaotic cryptosystems. In *Proc. IEEE Int. Carnahan Conf. Security Technology*, pages 332–338. IEEE, 1999.

[21] Marco Götz, Kristina Kelber, and Wolfgang Schwarz. Discrete-time chaotic encryption systems–Part I: Statistical design approach. *IEEE Trans. Circuits and Systems–I*, 44(10):963–970, 1997.

[22] Shujun Li, Xuanqin Mou, and Yuanlong Cai. Pseudo-random bit generator based on couple chaotic systems and its application in stream-ciphers cryptography. In *Progress in Cryptology – INDOCRYPT 2001*, Lecture Notes in Computer Science vol. 2247, pages 316–329. Springer-Verlag, Berlin, 2001.

[23] Andrzej Lasota and Michael C. Mackey. *Chaos, Fractals, and Noise - Stochastic Aspects of Dynamics*. Springer-Verlag, New York, second edition, 1997.

[24] Hong Zhou (周红), Jun Yu (俞军), and Xie-Ting Ling (凌燮亭). Design of chaotic feedforward stream cipher (混沌前馈型流密码的设计). *Acta Eletronica Sinica (电子学报)*, 26(1):98–101, 1998.

[25] Hong Zhou and Xie-Ting Ling. Problems with the chaotic inverse system encryption approach. *IEEE Trans. Circuits and Systems–I*, 44(3):268–271, 1997.

[26] Hong Zhou, Xie-Ting Ling, and Jie Yu. Secure communication via one-dimensional chaotic inverse systems. In *Proc. IEEE Int. Symposium Circuits and Systems 97*, volume 2, pages 9–12. IEEE, 1997.

[27] L. M. Pecora and T. L. Carroll. Synchronization in chaotic systems. *Physical Review Letters*, 64(8):821–824, 1990.

[28] T. Beth, D. E. Lazic, and A. Mathias. Cryptanalysis of cryptosystems based on remote chaos replication. In *Advances in Cryptology – EuroCrypt'94*, Lecture Notes in Computer Science vol. 0950, pages 318–331. Spinger-Verlag, Berlin, 1994.

[29] Kevin M. Short. Signal extraction from chaotic communications. *Int. J. Bifurcation and Chaos*, 7(7):1579–1597, 1997.

[30] Chang-Song Zhou and Tian-Lun Chen. Extracting information masked by chaos and contaminated with noise: Some considerations on the security of communication approaches using chaos. *Physics Letters A*, 234(6):429–435, 1997.

[31] Tao Yang, Lin-Bao Yang, and Chun-Mei Yang. Cryptanalyzing chaotic secure communications using return maps. *Physics Letters A*, 245(6):495–510, 1998.

[32] Maciej J. Ogorzatek and Hervé Dedieu. Some tools for attacking secure communication systems employing chaotic carriers. In *Proc. IEEE Int. Symposium Circuits and Systems 98*, volume 4, pages 522–525. IEEE, 1998.

[33] Stergios Papadimitriou, Anastasios Bezerianos, and Tassos Bounits. Radial basis function networks as chaotic generators for secure communication systems. *Int. J. Bifurcation and Chaos*, 9(1):221–232, 1999.

[34] Christopher P. Silva and Albert M. Young. Introduction to chaos-based communications and signal processing. In *Proc. IEEE Aerospace Conf.*, pages 279–299. IEEE, 2000.

[35] Andrew T. Parker and Kevin M. Short. Reconstructing the keystream from a chaotic encryption scheme. *IEEE Trans. Circuits and Systems–I*, 48(5):104–112, 2001.

[36] S. Papadimitriou, A. Bezerianos, T. Bounits, and G. Pavlides. Secure communication protocols with discrete nonlinear chaotic maps. *J. Systems Architechture*, 47(1):61–72, 2001.

[37] K. Murali, Haiyang Yu, Vinary Varadan, and Henry Leung. Secure communication using a chaos based signal encryption scheme. *IEEE Trans. Consuming Eletronics*, 47(4):709–714, 2001.

[38] Mohamed I. Sobhy and Alaa eldin R. Shehata. Chaotic algorithms for data encryption. In *2001 IEEE Int. Conf. Acoustics, Speech, and Signal Processing Proc. (ICASSP 2001)*, volume 2, pages 997–1000. IEEE, 2001.

[39] Mohamed I. Sobhy and Alaa-eldin R. Shehata. Methods of attacking chaotic encryption and countermeasures. In *2001 IEEE Int. Conf. Acoustics, Speech, and Signal Processing Proc. (ICASSP 2001)*, volume 2, pages 1001–1004. IEEE, 2001.

[40] Tung-Sheng Chiang and Peter Liu. Fuzzy model-based discrete-time chiang type chaotic cryptosystem. In *2001 IEEE Int. Conf. Fuzzy Systems Proc. (FUZZ-IEEE 2001)*, volume 3, pages 1404–1407. IEEE, 2001.

[41] Tung-Sheng Chiang, Chun-Chieh Wang, and Ching-Tsan Chiang. Robust T-S fuzzy model-based for chaotic cryptosystem. In *Proc. 2002 IEEE Int. Conf. Fuzzy Systems(FUZZ-IEEE'02)*, volume 1, pages 290–295. IEEE, 2002.

[42] Carlos Aguilar Ibáñez Hebert Sira-Ramírez and Miguel Suárez-Casta nón. Exact state reconstructors in the recovery of messages encrypted by the states of nonlinear discrete-time chaotic systems. *Int. J. Bifurcation and Chaos*, 12(1):169–177, 2002.

[43] Guojie Hu, Zhengjin Feng, and Ruiling Meng. Chosen ciphertext attack on chaos communication based on chaotic synchronization. *IEEE Trans. Circuits and Systems–I*, 50(2):275–279, 2003.

[44] Vladimir S. Udaltsov, Jean-Pierre Goedgebuer, Laurent Larger, Jean-Baptiste Cuenot, Pascal Levy, and William T. Rhodes. Cracking chaos-based encryption systems ruled by nonlinear time delay differential equations. *Physics Letters A*, 308(1):54–60, 2003.

[45] Roy Tenny, Lev S. Tsimring, Larry Larson, and Henry D. I. Abarbanel. Using distributed nonlinear dynamics for public key encryption. *Physical Review Letters*, 90(4):047903, 2003.

[46] P. G. Vaidya and Savita Angadi. Decoding chaotic cryptography without access to the superkey. *Chaos, Solitons & Fractals*, 17(2-3):379–386, 2003.

[47] Stephen Wolfram. Cryptography with cellular automata. In *Advances in Cryptology - Crypto'85*, Lecture Notes in Computer Science vol. 0218, pages 429–432. Spinger-Verlag, Berlin, 1985.

[48] Puhua Guan. Cellular automaton public-key cryptosystem. *Complex Systems*, 1:51–57, 1987.

[49] J.-P. Delahaye. Les automates (in French). *Pour la Science (French Edition of Scientific American)*, pages 126–134, Nov. 1991.

[50] H. A. Gutowitz. Cryptography with dynamical systems. In *Cellular Automata and Cooperative Phenomena*, Kluwer Academic Press, 1993.

[51] H. A. Gutowitz. Method and apparatus for encryption, decryption, and authentication using dynamical systems. US Patent No. 5365589, 1994.

[52] S. Nandi, B. K. Kar, and P. P. Chaudhuri. Theory and application of cellular automata in cryptography. *IEEE Trans. Computers*, 43(12):1346–1357, 1994.

[53] S. R. Blackburn, S. Murphy, and K. G. Paterson. Comments on "theory and application of cellular automata in cryptography". *IEEE Trans. Computers*, 46(5):637–638, 1997.

[54] S. Nandi and P. Pal Chaudhuri. Reply to comments on "theory and application of cellular automata in cryptography". *IEEE Trans. Computers*, 46(5):639, 1997.

[55] Jesús Uís, Edgardo Ugalde, and Gelasio Salazar. A cryptosystem based on cellular automata. *Chaos*, 8(4):819–822, 1998.

[56] N. Ganguly, A. Das, B. K. Sikdar, and P. P. Chaudhuri. Cellular automata model for cryptosystem. In *Proc. Cellular Automata Conference*, Yakohama National University, Japan, 2001.

[57] Subhayan Sen, Chandrama Shaw, Dipanwita Roy Chowdhuri, Niloy Ganguly, and P. Pal Chaudhuri. Cellular automata based cryptosystem (CAC). In *Information and Communications Security - 4th International Conference ICICS 2002 Proceedings*, Lecture Notes in Computer Science vol. 2513, pages 303–314. Spinger-Verlag, Berlin, 2002.

[58] Robert A. J. Matthews. On the derivation of a "chaotic" encryption algorithm. *Cryptologia*, XIII(1):29–42, 1989.

[59] Daniel D. Wheeler. Problems with chaotic cryptosystems. *Cryptologia*, XIII(3):243–250, 1989.

[60] Douglas W. Mitchell. Nonlinear key generators. *Cryptologia*, XIV(4):350–354, 1990.

[61] G. M. Bernstein and M. A. Lieberman. Secure random number generation using chaotic circuits. *IEEE Trans. Circuits and Systems*, 37(9):1157–1164, 1990.

[62] T. Habutsu, Y. Nishio, I. Sasase, and S. Mori. A secret key cryptosystem using a chaotic map. *Trans. IEICE*, E 73(7):1041–1044, 1990.

[63] T. Habutsu, Y. Nishio, I. Sasase, and S. Mori. A secret key cryptosystem by iterating a chaotic map. In *Advances in Cryptology – EuroCrypt'91*, Lecture Notes in Computer Science vol. 0547, pages 127–140. Spinger-Verlag, Berlin, 1991.

[64] Daniel D. Wheeler and Robert A. J. Matthews. Supercomputer investigations of a chaotic encryption algorithm. *Cryptologia*, XV(2):140–151, 1991.

[65] Daniel D. Wheeler. Problems with Mitchell's nonlinear key generators. *Cryptologia*, XV(4):355–151, 1991.

[66] E. Biham. Cryptoanalysis of the chaotic-map cryptosystem suggested at EuroCrypt'91. In *Advances in Cryptology – EuroCrypt'91*, Lecture Notes in Computer Science vol. 0547, pages 532–534. Spinger-Verlag, Berlin, 1991.

[67] R. Forré. The Hénon attractor as a keystream generator. In *Abstract of EuroCrypt'91 (wrong?)*, pages 76–80, 1991. (This paper is cited in [70] with wrong source).

[68] G. M. Berstein and M. A. Lieberman. Method and apparatus for generating secure random numbers using chaos. US Patent No. 5007087, 1991.

[69] M. E. Bianco and D. A. Reed. Encryption system based on chaos theory. US Patent No. 5048086, 1991.

[70] D. Erdmann and S. Murphy. Hénon stream cipher. *Electronics Letters*, 28(9):893–895, 1992.

[71] Ross Anderson. Letter to the editor: Chaos and random numbers. *Cryptologia*, XVI(3):226, 1992.

[72] Fengi Hwu. *The Interpolating Random Spline Cryptosystem and the Chaotic-Map Public-Key Cryptosystem*. PhD thesis, Faculty of the Graduate School, University of Missouri - Rolla, 1993.

[73] D. R. Frey. Chaotic digital encoding: An approach to secure communication. *IEEE Trans. Circuits and Systems–II*, 40(10):660–666, 1993.

[74] V. A. Protopopescu, R. T. Santoro, and J. S. Tollover. Fast and secure encryption – decryption method based on chaotic dynamics. US Patent No. 5479513, 1995.

[75] Tohru Kohda and Akio Tsuneda. Chaotic bit sequences for stream cipher cryptography and their correlation functions. In *Chaotic Circuits for Communication*, Proceedings of SPIE vol. 2612, pages 86–97, 1995.

[76] Ute Feldmann, Martin Hasler, and Wolfgang Schwarz. Communication by chaotic signals: The inverse system approach. *Int. J. Circuit Theory and Applications*, 24(5):551–579, 1996.

[77] Hong Zhou (周红). *A Design Methodology of Chaotic Stream Ciphers and the Realization Problems in Finite Precision* (《一类混沌密码序列的设计方法及其有限精度实现问题分析》). PhD thesis, Department of Electronic Engineering, Fudan University (复旦大学电子工程系), Shanghai, China, June 1996.

[78] Hong Zhou (周红), Jie Luo (罗杰), and Xie-Ting Ling (凌燮亭). Generating nonlinear feedback stream ciphers via chaotic systems (混沌非线性反馈密码序列的理论设计和有限精度实现). *Acta Eletronica Sinica* (电子学报), 25(10):57–60,56, 1997.

[79] Hong Zhou and Xieting Ling. Generating chaotic secure sequences with desired statistical properties and high security. *Int. J. Bifurcation and Chaos*, 7(1):205–213, 1997.

[80] Zbigniew Kotulski and Janusz Szczepanski. Discrete chaotic cryptography. *Annalen der Physik*, 6(5):381–394, 1997.

[81] Tao Sang, Ruili Wang, and Yixun Yan. Perturbance-based algorithm to expand cycle length of chaotic key stream. *Electronics Letters*, 34(9):873–874, 1998.

[82] Tao Sang, Ruili Wang, and Yixun Yan. Clock-controlled chaotic keystream generators. *Electronics Letters*, 34(20):1932–1934, 1998.

[83] Frank Dachselt, Kristina Kelber, and Wolfgang Schwarz. Discrete-time chaotic encryption systems–Part III: Cryptographical analysis. *IEEE Trans. Circuits and Systems–I*, 45(9):983–988, 1998.

[84] M. S. Baptista. Cryptography with chaos. *Physics Letters A*, 240(1-2):50–54, 1998.

[85] Josef Scharinger. Fast encryption of image data using chaotic Kolmogrov flows. *J. Electronic Imaging*, 7(2):318–325, 1998.

[86] Christopher F. Woodcock and Nigel P. Smart. *p*-adic chaos and random number generation. *Experimental Mathematics*, 7(4):333–342, 1998.

[87] Donghui Guo, L. M. Cheng, and L. L. Cheng. A new symmetric probabilistic encryption scheme based on chaotic attractors of neural networks. *Applied Intelligence*, 10(1):71–84, 1999.

[88] W. G. Chambers. Comments on "Chaotic digital encoding: An approach to secure communication". *IEEE Trans. Circuits and Systems–II*, 46(11):1445–1447, 1999.

[89] Masaki Miyamoto, Kiyoshi Tanaka, and Tatsuo Sugimura. Truncated Baker transformation and its extension to image encryption. In *Mathematics of Data/Image Coding, Compression, and Encryption II*, Proceedings of SPIE vol. 3814, pages 13–25, 1999.

[90] E. Alvarez, A. Fernández, P. García, J. Jiménez, and A. Marcano. New approach to chaotic encryption. *Physics Letters A*, 263(4-6):373–375, 1999.

[91] Zbigniew Kotulski and Janusz Szczepanski. Application of discrete chaotic dynamical systems in cryptography – DCC method. *Int. J. Bifurcation and Chaos*, 9(6):1121–1135, 1999.

[92] Tao Sang (桑涛), Ruili Wang (王汝笠), and Yixun Yan (严义埙). The theoretical design for a class of new chaotic feedback stream ciphers (一类新型混沌反馈密码序列的理论设计). *Acta Eletronica Sinica (电子学报)*, 27(7):47–50, 1999.

[93] Toru Ohira. Encryption with delayed dynamics. *Computer Physics Communications*, 121-122:75–82, 1999.

[94] Dong-Hui Guo (郭东辉), Xiao-Juan He (何小娟), and Cai-Sheng Chen (陈彩生). ASIC design of chaotic encryption based on neural networks (基于神经网络混沌加密算法的专用芯片设计). *Chinese J. Computers (计算机学报)*, 23(11):1230–1232, 2000.

[95] F. Argenti, S. Benzi, E. Del Re, and R. Genesio. Stream cipher system based on chaotic maps. In *Mathematics and Applications of Data/Image Coding, Compression, and Encryption III*, Proceedings of SPIE vol. 4122, pages 10–17, 2001.

[96] Mieczysław Jessa. Data encryption algorithms using one-dimensional chaotic maps. In *Proc. IEEE Int. Symposium Circuits and Systems 2000*, volume I, pages 711–714. IEEE, 2000.

[97] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Cryptanalysis of a chaotic encryption system. *Physics Letters A*, 276(1-4):191–196, 2000.

[98] Li-Hui Zhou and Zheng-Jin Feng. A new idea of using one-dimensional PWL map in digital secure communications–dual-resolution approach. *IEEE Trans. Circuits and Systems–II*, 47(10):1107–1111, 2000.

[99] Ninan Sajeeth Philip and K. Babu Joseph. Chaos for stream cipher. arXiv:nLin.CD/0102012 v1, 16 Feb. 2001, available online at http://arxiv.org/abs/cs.CR/0102012.

[100] Goce Jakimoski and Ljupčo Kocarev. Analysis of some recently proposed chaos-based encryption algorithms. *Physics Letters A*, 291(6):381–384, 2001.

[101] Ljupčo Kocarev. Chaos-based cryptography: A brief overview. *IEEE Circuits and Systems Maganize*, 1(3):6–21, 2001.

[102] Frank Dachselt and Wolfgang Schwarz. Chaos and cryptography. *IEEE Trans. Circuits and Systems–I*, 48(12):1498–1509, 2001.

[103] Roland Schmitz. Use of chaotic dynamical systems in cryptography. *J. Franklin Institute*, 338(4):429–441, 2001.

[104] Wai-Kit Wong, Lap-Piu Lee, and Kwok-Wo Wong. A modified chaotic cryptographic method. *Computer Physics Communications*, 138(3):234–236, 2001.

[105] Ljupčo Kocarev and Goce Jakimoski. Logistic map as a block encryption algorithm. *Physics Letters A*, 289(4-5):199–206, 2001.

[106] Stergios Papadimitriou, Tassos Bountis, Seferina Mavaroudi, and Anastassions Bezerianos. A probabilistic symmetric encryption scheme for very fast secure communications based on chaotic systems of difference equations. *Int. J. Bifurcation and Chaos*, 11(12):3107–3115, 2001.

[107] Naoki Masuda and Kazuyuki Aihara. Cryptosystems based on space-discretization of chaotic maps. In *Proc. IEEE Int. Symposium Circuits and Systems 2001*, volume III, pages 321–324. IEEE, 2001.

[108] Goce Jakimoski and Ljupčo Kocarev. Chaos and cryptography: Block encryption ciphers based on chaotic maps. *IEEE Trans. Circuits and Systems–I*, 48(2):163–169, 2001.

[109] Shujun Li, Qi Li, Wenmin Li, Xuanqin Mou, and Yuanlong Cai. Statistical properties of digital piecewise linear chaotic maps and their roles in cryptography and pseudo-random coding. In *Cryptography and Coding – 8th IMA Int. Conf. Proc.*, Lecture Notes in Computer Science vol. 2260, pages 205–221. Springer-Verlag, Berlin, 2001.

[110] Shujun Li, Xuanqin Mou, and Yuanlong Cai. Improving security of a chaotic encryption approach. *Physics Letters A*, 290(3-4):127–133, 2001.

[111] GuoJie Hu, ZhengJin Feng, and Lin Wang. Analysis of a type digital chaotic cryptosystem. In *Proc. IEEE Int. Symposium Circuits and Systems 2002*, volume III, pages 473–475. IEEE, 2002.

[112] Shujun Li, Xuan Zheng, Xuanqin Mou, and Yuanlong Cai. Chaotic encryption scheme for real-time digital video. In *Real-Time Imaging VI*, Proceedings of SPIE vol. 4666, pages 149–160, 2002.

[113] A. Palacios and H. Juarez. Cryptography with cycling chaos. *Physics Letters A*, 303(5-6):345–351, 2002.

[114] Kwok-Wo Wong. A fast chaotic cryptographic scheme with dynamic look-up table. *Physics Letters A*, 298(4):238–242, 2002.

[115] P. García, A. Parravano, M. G. Cosenza, J. Jiménez, and A. Marcano. Coupled map networks as communication schemes. *Physical Review E*, 65(4):045201(R), 2002.

[116] P. García and J. Jiménez. Communication through chaotic map systems. *Physics Letters A*, 298(1):34–40, 2002.

[117] Naoki Masuda and Kazuyuki Aihara. Cryptosystems with discretized chaotic maps. *IEEE Trans. Circuits and Systems–I*, 49(1):28–40, 2002.

[118] Xun Yi, Chik How Tan, and Chee Kheong Siew. A new block cipher based on chaotic tent maps. *IEEE Trans. Circuits and Systems–I*, 49(12):1826–1829, 2002.

[119] Shihong Wang, Jinyu Kuang, Jinghua Li, Yunlun Luo, Huaping Lu, and Gang Hu. Chaos-based secure communications in a large community. *Physical Review E*, 66(6):065202(R), 2002.

[120] Mieczysław Jessa. Data transmission with adjustable security exploiting chaos-based pseudorandom number generators. In *Proc. IEEE Int. Symposium Circuits and Systems 2002*, volume III, pages 476–479. IEEE, 2002.

[121] Goce Jakimoski and Ljupčo Kocarev. Differenytial and linear probabilities of a block-encryption cipher. *IEEE Trans. Circuits and Systems–I*, 50(1):121–123, 2003.

[122] Kwok-Wo Wong. A combined chaotic cryptographic and hashing scheme. *Physics Letters A*, 307(5-6):292–298, 2003.

[123] Kwok-Wo Wong, Sun-Wah Ho, and Ching-Ki Yung. A chaotic cryptography scheme for generating short ciphertext. *Physics Letters A*, 310(1):67–73, 2003.

[124] Shihong Wang, Weiping Ye, Huaping Lu, Jinyu Kuang, Jinghua Li, Yunlun Luo, and Gang Hu. A spatiotemporal-chaos-based encryption having overall properties considerably better than advanced encryption standard. arXiv:nLin.CD/0303026 v1, 14 Mar. 2003, available online at http://arxiv.org/abs/nlin.CD/0303026.

[125] N. K. Pareek, Vinod Patidar, and K. K. Sud. Discrete chaotic cryptography using external key. *Physics Letters A*, 309(1-2):75–82, 2003.

[126] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Cryptanalysis of an ergodic chaotic cipher. *Physics Letters A*, 311(2-3):172–179, 2003.

[127] G. Alvarez, F. Montoya, M. Romera, and G. Pastor. Crytpanalysis of a chaotic secure communication system. *Physics Letters A*, 306(4):200–205, 2003.

[128] Shujun Li, Xuanqin Mou, Zhen Ji, Jihong Zhang, and Yuanlong Cai. Performance analysis of Jakimoski-Kocarev attack on a class of chaotic cryptosystems. *Physics Letters A*, 307(1):22–28, 2003.

[129] Shujun Li, Xuanqin Mou, Yuanlong Cai, Zhen Ji, and Jihong Zhang. On the security of a chaotic encryption scheme: Problems with computerized chaos in finite computing precision. *Computer Physics Communications*, 153(1):52–58, 2003.

[130] Shujun Li, Xuanqin Mou, Boliya L. Yang, Zhen Ji, and Jihong Zhang. Problems with a probabilistic encryption scheme based on chaotic systems. *Int. J. Bifurcation and Chaos*, 13(10):3063–3077, 2003.

[131] Shujun Li, Xuanqin Mou, Luhua Gong, and Yuanlong Cai. On the security of a chaotic cipher to Biham's attacks. unpublished, 2002.

[132] Jui-Cheng Yen and Jiun-In Guo. A new image encryption algorithm and its VLSI architecture. In *Proc. IEEE Workshop Signal Processing Systems*, pages 430–437, 1999.

[133] Scott Su, Alvin Lin, and Jui-Cheng Yen. Design and realization of a new chaotic neural encryption/decryption network. In *Proc. 2000 IEEE Asia-Pacific Conf. Circuits and Systems (APCCAS 2000)*, pages 335–338. IEEE, 2000.

[134] Jui-Cheng Yen and Jiun-In Guo. A new chaotic key-based design for image encryption and decryption. In *Proc. IEEE Int. Symposium Circuits and Systems 2000*, volume 4, pages 49–52, 2000.

[135] Jui-Cheng Yen and Jiun-In Guo. Efficient hierarchical chaotic image encryption algorithm and its vlsi realisation. *IEE Proc.-Vis. Image Signal Process.*, 147(2):167–175, 2000.

[136] Kenji Yano and Kiyoshi Tanaka. Image encryption scheme based on a truncated Baker transformation. *IEICE Trans. Fundamentals*, E85-A(9):2025–2035, 2002.

[137] Jui-Cheng Yen and Jiun-In Guo. Design of a new signal security system. In *Proc. IEEE Int. Symposium Circuits and Systems 2002*, volume IV, pages 121–124. IEEE, 2002.

[138] Jiun-In Guo, Jui-Cheng Yen, and H.-F. Pan. New voice over Internet protocol technique with hierarchical data security protection. *IEE Proc.-Vis. Image Signal Process.*, 149(4):237–243, 2002.

[139] Shujun Li and Xuan Zheng. Cryptanalysis of a chaotic image encryption method. In *2002 IEEE Int. Sym. Circuits and Systems Proc. (ISCAS 2002)*, pages 708–711, 2002.

[140] Shujun Li and Xuan Zheng. On the security of an image encryption method. In *Proc. 2002 Int. Conf. Image Processing (ICIP 2002)*, volume 2, pages 925–928, 2002.

[141] Shujun Li (李树钧), Xuanqin Mou (牟轩沁), Zhen Ji (纪震), and Jihong Zhang (张基宏). Cryptanalysis of a class of chaotic stream ciphers (一类混沌流密码的分析). *Journal of Electronics & Information Technology (电子与信息学报)*, 25(4):473–478, 2003.

[142] C. E. Shanon. Communication theory of secrecy systems. *Bell Sys. Tech. J.*, 28(4):656–715, 1949.

[143] B. V. Chirkikov and F. Vivaldi. An algorithmic view of pseudochaos. *Physica D*, 129(3-4):223–235, 1999.

[144] Bruce Schneier. *Applied Cryptography – Protocols, algorithms, and souce code in C*. John Wiley & Sons, Inc., New York, second edition, 1996.

[145] Yumin Wang (王育民) and Jianwei Liu (刘建伟). *Security of Communication Networks: Theory and Techniques (《通信网的安全－理论与技术》)*. Xidian University Press (西安电子科技大学出版社), Xi'an, China, 1999.

[146] Zhen Ji (纪震). *Research on Key Techniques in Medical DSA Imaging Systems (《医用DSA系统的关键技术研究》)*. PhD thesis, School of Electronics and Information Engineering, Xi'an Jiaotong University (西安交通大学电子与信息工程学院), Xi'an, China, March 1999.

[147] Shin'ichi Oishi and Hajime Inoue. Pseudo-random number generators and chaos. *Trans. IECE Japan*, E 65(9):534–541, 1982.

[148] F. James. A review of pseudorandom number generators. *Computer Physics Communications*, 60(3):329–344, 1990.

[149] Ghobad Heidari-Bateni and Clare D. McGillem. A chaotic direct-sequence spread-spectrum communication system. *IEEE Trans. Communications*, 42(2/3/4):1524–1527, 1994.

[150] S. C. Phatak and S. Suresh Rao. Logistic map: A possible random-number generator. *Physical Review E*, 51(4):3670–3678, 1995.

[151] Tohru Kohda and Akio Tsuneda. Statistics of chaotic binary sequences. *IEEE Trans. Information Theory*, 43(1):104–112, 1997.

[152] Gianluca Mazzini, Gianluca Setti, and Riccardo Rovatti. Chaotic complex spreading spectrum sequences for asynchronous DS-CDMA–Part I: System modeling and results. *IEEE Trans. Circuits and Systems–I*, 44(10):937–947, 1997.

[153] Riccardo Rovatti, Gianluca Mazzini, and Gianluca Setti. Chaotic complex spreading spectrum sequences for asynchronous DS-CDMA–Part II: Some theoretical performance bounds. *IEEE Trans. Circuits and Systems–I*, 45(4):496–506, 1998.

[154] Jorge A. González and Ramiro Pino. A random number generator based on unpredictable chaotic functions. *Computer Physics Communications*, 120(2-3):109–114, 1999.

[155] Jorge A. Gonzalez, Miguel Martin-Landrove, and Leonardo Trujillo. Absolutely unpredictable chaotic sequences. *Int. J. Bifurcation and Chaos*, 10(8):1867–1874, 2000.

[156] Ling Cong and Li Shaoqian. Chaotic spreading sequences with multiple access performance better than random sequences. *IEEE Trans. Circuits and Systems–I*, 47(3):394–397, 2000.

[157] Agner Fog. Chaotic random number generators with random cycle lengths. downloadable at http://www.agner.org/random/theory/chaosran.doc, Nov. 2001.

[158] R. Bernardini and G. Cortelazzo. Tools for designing chaotic systems for secure random number generation. *IEEE Trans. Circuits and Systems–I*, 48(5):552–564, 2001.

[159] Janusz Szczepański and Zbigniew Kotulski. Pseudorandom number generators based on chaotic dynamical systems. *Open Sys. & Information Dyn.*, 8(2):137–146, 2001.

[160] Toni Stojanovski and Ljupčo Kocarev. Chaos-based random number generators–Part I: Analysis. *IEEE Trans. Circuits and Systems–I*, 48(3):281–288, 2001.

[161] Toni Stojanovski, Johnny Pihl, and Ljupčo Kocarev. Chaos-based random number generators–Part II: Practical realization. *IEEE Trans. Circuits and Systems–I*, 48(3):382–385, 2001.

[162] M. Jessa. The period of sequences generated by tent-like maps. *IEEE Trans. Circuits and Systems–I*, 49(1):84–89, 2002.

[163] Mieczysław Jessa and Marcin Walentynowicz. Discrete-time phase-locked loop as a source of random sequences with different distributions. In *Proc. IEEE Int. Symposium Circuits and Systems 2002*, volume III, pages 189–192. IEEE, 2002.

[164] Ljupčo Kocarev and Goce Jakimoski. Pseudorandom bits generated by chaotic maps. *IEEE Trans. Circuits and Systems–I*, 50(1):123–126, 2003.

[165] Hongtao Zhang, Huiyun Wang, and Wai-Kai Chen. Oversampled chaotic binary sequences with good security. *J. Circuits, Systems and Computers*, 11(2):173–185, 2002.

[166] IEEE Computer Society. Advanced encryption standard (AES). Federal Information Processing Standards Publication 197 (FIPS-197), downloadable at http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf, November 2001.

[167] Terry Ritter. Substitution cipher with pseudo-random shuffling: The dynamic substitution combiner. *Cryptologia*, XIV(4):289–303, 1990.

[168] Terry Ritter. Transposition cipher with pseudo-random shuffling: The dynamic transposition combiner. *Cryptologia*, XV(1):1–17, 1991.

[169] Donald E. Knuth. *The Art of Computer Programing Volume 2: Seminumerical Algorithms*. Addison-Wesley, third edition, 1998.

[170] Hong Zhou (周红) and Xieting Ling (凌燮亭). Realizing finite precision chaotic systems via perturbation of *m*-sequences (有限精度混沌系统的*m*序列扰动实现). *Acta Eletronica Sinica (电子学报)*, 25(7):95–97, 1997.

[171] Naoki Masuda and Kazuyuki Aihara. Dynamical characterstics of discretized chaotic permutations. *Int. J. Bifurcation and Chaos*, 12(10):2087–2103, 2002.

[172] Michael Blank. *Discreteness and Continuity in Problems of Chaotic Dynamics*, volume 161 of *Translations of Mathematical Monographs*. American Mathematical Society, Providence, Rhode Island, 1997.

[173] Y. E. Levy. Some remarks about computer studies of dynamical systems. *Physics Letters A*, 88(1):1–3, 1982.

[174] F. Rannou. Numerical study of discrete plane area-preserving mappings. *Astronomy and Astrophysics*, 31:289–301, 1974.

[175] G. Benettin, M. Casartelli, L. Galgani, A. Giorgilli, and J.-M. Strekcyn. On the reliability of numerical studies of stochasticity I: Exsitence of time average. *IL Nuovo Cimento B*, 44(1):183–195, 1978.

[176] Charles F. F. Karney. Long-time correlations in the stochastic regime. *Physica D*, 8(3):360–380, 1983.

[177] T. Hogg and B. A. Huberman. Attractors on finite sets: The dissipative dynamics of computing structures. *Physical Review A*, 32(4):2338–2346, 1985.

[178] W. F. Wolffand B. A. Huberman. Transients and asymptotics in granular phase space. *Zeitschrift für Physik B - Condensed Matter*, 63:397–405, 1986.

[179] P. M. Binder and R. V. Jensen. Simulating chaotic behavior with finite-state machines. *Physical Review A*, 34:4460–4463, 1986.

[180] Joseph L. McCauley and Julian I. Palmore. Computable chaotic orbits. *Physics Letters A*, 115(9):433–436, 1986.

[181] Julian I. Palmore and Joseph L. McCauley. Shadowing by computable chaotic orbits. *Physics Letters A*, 122(8):399–402, 1987.

[182] Ian Percival and Franco Vivaldi. Arithmetical properties of strongly chaotic maps. *Physica D*, 25(1-3):105–130, 1987.

[183] C. Beck and G. Roepstorff. Effects of phase space discretization on the long-time behavior of dynamical systems. *Physica D*, 25(1-3):95–97, 1987.

[184] Kunihiko Kaneko. Symplectic cellular automata. *Physics Letters A*, 129(1):9–16, 1988.

[185] Celso Grebogi, Edward Ott, and James A. Yorke. Roundoff-induced periodicity and the correlation dimension of chaotic attractors. *Physical Review A*, 38(7):3688–3692, 1988.

[186] P. Góra and A. Boyarsku. Why computers like Lebesgue measure. *Computers & Mathematics with Applications*, 16(4):321–329, 1988.

[187] E. Thiran, D. Verstegen, and J. Weyers. p-adic dynamics. *J. Statistical Physics*, 54(3-4):893–913, 1989.

[188] Julian Palmore and Charles Herring. Computer arithmetic, chaos and fractals. *Physica D*, 42(1-3):99–110, 1990.

[189] J. P. Keating. Asmptoic properties of the periodic orbits of the cat maps. *Nonlinearity*, 4(2):277–307, 1991.

[190] Slawomir T. Fryska and Mohamed A. Zohdy. Computer dynamics and shadowing of chaotic orbits. *Physics Letters A*, 166(5-6):340–346, 1992.

[191] P.-M. Binder. Limit cycles in a quadratic discrete iteration. *Physica D*, 57(1-2):31–38, 1992.

[192] David J. D. Earn and Scott Tremaine. Exact numerical studies of hamiltonian maps: Iterating without roundoff error. *Physica D*, 56(1):1–22, 1992.

[193] P. H. Borcherds and G. P. McCauley. The digital tent map and the trapezoidal map. *Chaos, Solitons & Fractals*, 3(4):451–466, 1993.

[194] P. Diamond, P. Kloeden, and A. Pokrovskii. An invariant measure arising in computer simulation of a chaotic dynamical system. *J. Nonlinear Science*, 4:59–68, 1994.

[195] Michael Blank. Pathologies generated by round-off in dynamical systems. *Physica D*, 78(1-2):93–114, 1994.

[196] F. Vivaldi. Periodicity and transport from round-off errors. *Experimental Mathematics*, 3(4):303–315, 1994.

[197] David K. Arrowsmith and F. Vivaldi. Geometry of p-adic Siegel discs. *Physica D*, 71(1-2):222–236, 1994.

[198] P. Diamond, P. Kloeden, A. Pokrovskii, and A. Vladimirov. Collapsing effects in numerical simulation of a class of chaotic dynamical systems and random mappings with a single attracting centre. *Physica D*, 86(4):559–571, 1995.

[199] J. Čermák. Digital generators of chaos. *Physics Letters A*, 214(3-4):151–160, 1996.

[200] J. H. Lowenstein and F. Vivaldi. Anomalous trasport in a model of Hamiltonian round-off. *Nonlinearity*, 11(5):1321–1350, 1998.

[201] Xu-Sheng Zhang and F. Vivaldi. Small perturbations of a discrete twist map. *Physique Theorique*, 68(4):507–523, 1998.

[202] Henri Waelbroeck and Federico Zertuche. Discrete chaos. *J. Physics A*, 32(1):175–189, 1999.

[203] A. V. Pokrovskii, A. Kent, and J. McInerney. Mixed moments of random mappings and chaotic dynamical systems. Technical Report Report 99-003, Institute for Nonlinear Science (INS) at UCC, University College, Cork, Ireland, March 1999.

[204] W. G. Chambers. Orbit-periods in second-order finite-precision digital filters with overflow. *Int. J. Bifurcation and Chaos*, 9(8):1669–1674, 1999.

[205] D. Bosioand and F. Vivaldi. Round-off errors and p-adic numbers. *Nonlinearity*, 13(1):309–322, 2000.

[206] Shi-Gang Chen (陈式刚). *Maps & Chaos (《映象与混沌》)*. National Defense Industry Press (国防工业出版社), Beijing, China, 1992.

[207] Francois Robert. *Discrete Iterations: A Metric Study*. Springer Series in Computational Mathematics vol. 6. Springer-Verlag, Berlin, 1986.

[208] Weimin Zheng (郑维敏). *Positive Feedback (《正反馈》)*. Tsinghua University Press (清华大学出版社), Beijing, China, 1998.

[209] Bai-Lin Hao (郝柏林). *Starting with Parabolas: An Introduction to Chaotic Dynamics (《从抛物线谈起：混沌动力学引论》)*. Shanghai Scientific and Technological Education Publishing House (上海科技教育出版社), Shanghai, China, 1993.

[210] A. Baranovsky and D. Daems. Design of one-dimensional chaotic maps with prescribed statistical properties. *Int. J. Bifurcation and Chaos*, 5(6):1585–1598, 1995.

[211] Guanzhang Hu (胡冠章). *Applied Modern Algebra (《应用近世代数》)*. Tsinghua University Press (清华大学出版社), Beijing, China, second edition, 1999.

[212] The Editorial Committee of *Modern Applied Mathematics Handbook* (《现代应用数学手册》编委会). *Modern Applied Mathematics Handbook – vol. Probability Theory and Stochastic Process (《现代应用数学手册：概率论与随机过程卷》)*. Tsinghua University Press (清华大学出版社), Beijing, China, 2000.

[213] Cunsheng Ding (丁存生) and Guozhen Xiao (肖国镇). *Stream-Cipher Cryptology and Its Applications (《流密码学及其应用》)*. National Defense Industry Press (国防工业出版社), Beijing, China, 1994.

[214] Guozhen Xiao (肖国镇) and Yumin Wang (王育民). *Pseudo-Random Sequences and its Applications (《伪随机序列及其应用》)*. National Defense Industry Press (国防工业出版社), Beijing, China, 1985.

[215] Shi-Hua Chen (陈士华) and Jun-An Lu (陆君安). *Introduction to Chaos Dynamics (《混沌动力学初步》)*. Wuhan Water Resource and Electric Power University Press (武汉水力水电大学出版社), Wuhan, China, 1998.

[216] Kenneth R. Castleman. *Digital Image Processing*. Prentice Hall Inc., New York, 1996.

[217] IEEE Computer Society. IEEE standard for binary floating-point arithmetic. ANSI/IEEE Std. 754-1985, August 1985.

[218] Doug Stinson. Visual cryptography & threshold schemes. *Dr. Dobb's J. Software Tools for Professional Programmer*, 23(4):36, 38–43, April 1998.

[219] Stergios Papadimitriou, Anastasios Bezerianos, and Tassos Bounits. Secure communication with chaotic systems of difference equations. *IEEE Trans. Computers*, 46(1):27–38, 1997.

[220] Zhen-Sheng Yang (杨振生). *Combinatorial Mathematics and Algorithms (《组合数学及其算法》)*. University of Science and Technology of China Press (中国科技大学出版社), Hefei, China, 1997.

[221] Howard Cheng and Xiaobo Li. Partial encryption of compressed images and videos. *IEEE Trans. Signal Processing*, 48(8):2439–2451, 2000.

[222] Philip P. Dang and Paul M. Chau. Image encryption for secure Internet multimedia applications. *IEEE Trans. Consumer Electronics*, 46(3):395–403, 2000.

[223] Henry Ker-Chang Chang and Jiang-Long Liu. A linear quadtree compression scheme for image encryption. *Signal Processing: Image Communication*, 10(4):279–290, 1997.

[224] C. Alexopoulos, Nikolaos G. Bourbakis, and N. Ioannou. Image encrytion method using a class of fractals. *J. Electronic Imaging*, 4(3):251–259, 1995.

[225] N. Bourbakis and C. Alexopoulos. Picture data encryption using scan patterns. *Pattern Recognition*, 25(6):567–581, 1992.

[226] Jinn-Ke Jan and Yuh-Min Tseng. On the security of image encryption method. *Information Processing Letters*, 60(5):261–265, 1996.

[227] Yixian Yang (杨义先) and Xuduan Lin (林须端). *Coding Theory and Cryptology (《编码密码学》)*. People's Post and Telecommunications Press (人民邮电出版社), Beijing, China, 1992.

[228] Ali Şaman Tosun. Lightweight security mechanisms for wireless video transmission. In *Proc. Int. Conf. Information Technology*, pages 157–161, 2001.

[229] Sang Uk Shin, Kyeong Seop Sim, and Kyung Hyune Rhee. A secrecy scheme for MPEG video data using the joint of compression and encryption. In *Proc. Int. Information Security Workshop*, Lecture Notes in Computer Science **1729**, pages 191–201, Berlin, 1999. Springer-Verlag.

[230] Lintian Qiao and Klara Nahrstedt. A new algorithm for MPEG video encryption. In *Proc. Int. Conf. Imaging Science, Systems, and Technology*, pages 21–29, 1997.

[231] Xiaolin Wu and Peter W. Moo. Joint image/video compression and encryption via high-order conditional entropy coding of wavelet coefficients. In *Proc. Int. Conf. Multimedia Computing and Systems*, pages 908–912, 1999.

[232] Changgui Shi, Sheng-Yih Wang, and Bharat Bhargava. MPEG video encryption in real-time using secret key cryptography. In *Proc. Int. Conf. Parallel and Distributed Processing Techniques and Applications*, 1999.

[233] Changgui Shi and Bharat Bhargava. A fast MPEG video encryption algorithm. In *Proc. ACM Multimedia 98*, pages 81–88, 1998.

[234] Lei Tang. Methods for encrypting and decrypting MPEG video data efficiently. In *Proc. ACM Multimedia 96*, pages 219–230, 1996.

[235] Iskender Agi and Li Gong. An empirical study of secure MPEG video transmissions. In *Proc. Internet Society Sym. Network and Distributed Systems Security*, pages 137–144, 1996.

[236] Yongcheng Li, Zhigang Chen, See-Mong Tan, and Roy H. Campbell. Security enhanced MPEG player. In *Proc. Int. Workshop Multimedia Software Development*, pages 169–175, 1996.

[237] George Anastasios Spanos and Tracy Bradley Maples. Performance study of a selective encryption scheme for the security of networked, real-time video. In *Proc. Int. Conf. Computer Communications and Networks*, pages 2–10, 1995.

[238] Lintian Qiao and Klara Nahrstedt. Comparison of MPEG encryption algorithms. *Computers & Graphics*, 22(4):437–448, 1998.

[239] Lintian Qiao, Klara Nahrstedt, and Ming-Chit Tam. Is MPEG encryption by using random list instead of Zig-Zag order secure? In *Proc. Int. Sym. Consumer Electronics*, pages 226–229, 1997.

[240] Takeyuki Uehara and Reihaneh Safavi-Naini. Chosen DCT coefficients attack on MPEG encryption schemes. In *Proc. IEEE Pacific Rim Conf. Multimedia*, pages 316–319, 2000.

[241] Thomas Sikora. MPEG digital video-coding standards. *IEEE Signal Processing Maganize*, 14(5):82–100, 1997.

[242] Tohru Kohda and Kazuyuki Aihara. Chaos in discrete systems and diagnosis of experimental chaos. *Trans. IEICE*, E 73(6):772–783, 1990.

My Publications Related to this Thesis (攻读博士期间发表相关文章列表)

[1] **Shujun Li**, Xuanqin Mou, and Yuanlong Cai. Improving security of a chaotic encryption approach. *Physics Letters A*, 290(3-4):127–133, 2001. (**SCI** indexed, IDS Number: 495EE).

[2] **Li Shujun**, Mou Xuanqin, and Cai Yuanlong. Pseudo-random bit generator based on couple chaotic systems and its application in stream-ciphers cryptography. In *Progress in Cryptology - INDOCRYPT 2001*, Lecture Notes in Computer Science vol. 2247, pages 316–329. Springer-Verlag, Dec. 2001. (**SCI Expanded** Source).

[3] **Shujun Li**, Qi Li, Wenmin Li, Xuanqin Mou, and Yuanlong Cai. Statistical properties of digital piecewise linear chaotic maps and their roles in cryptography and pseudo-random coding. In *Cryptography and Coding–8th IMA International Conference Proceedings*, Lecture Notes in Computer Science vol. 2260, pages 205–221. Springer-Verlag, Dec. 2001. (**SCI Expanded** Source).

[4] **Shujun Li**, Xuan Zheng, Xuanqin Mou, and Yuanlong Cai. Chaotic encryption scheme for real-time digital video. In *Real-Time Imaging VI*, Proceedings of SPIE vol. 4666, pages 149–160, 2002. (**EI** indexed, AN: 7203934, New AN: 02467203934; **ISTP** indexed, IDS Number: BU44J).

[5] **Shujun Li** and Xuan Zheng. Cryptanalysis of a chaotic image encryption method. In *Proceedings of 2002 IEEE International Symposium on Circuits and Systems (ISCAS 2002)*, pages 708–711, 2002. (**EI** indexed, AN: 7011208, New AN: 02287011208).

[6] **Shujun Li** and Xuan Zheng. On the security of an image encryption method. In *Proceedings of 2002 IEEE International Conference on Image Processing (ICIP 2002)*, volume 2, pages 925–928, 2002. (**EI** indexed, AN: 7288882, New AN: 02517288882).

[7] **Shujun Li**, Xuanqin Mou, Zhen Ji, Jihong Zhang, and Yuanlong Cai. Performance analysis of Jakimoski-Kocarev attack on a class of chaotic cryptosystems. *Physics Letters A*, 307(1):22–28, 2003. (**SCI** indexed, IDS Number: 639BJ, an erraum of this paper has been published in *Physics Letters A*, vol. 309, no. 1-2, pp. 165, **SCI** indexed, IDS Number: 656BM).

[8] **Shujun Li (李树钧)**, Xuanqin Mou (牟轩沁), Zhen Ji (纪震), and Jihong Zhang (张基宏). Cryptanalysis of a class of chaotic stream ciphers (一类混沌流密

码的分析). *Journal of Electronics & Information Technology (电子与信息学报)*, 25(4):473–478, 2003. (**EI** indexed, AN: 7499784, New AN: 03237499784).

[9] **Shujun Li**, Xuanqin Mou, Yuanlong Cai, Zhen Ji, and Jihong Zhang. On the security of a chaotic encryption scheme: Problems with computerized chaos in finite computing precision. *Computer Physics Communications*, 153(1):52–58, 2003. (**SCI** indexed, IDS Number: 683VL; **EI** indexed, AN: 7489635, New AN: 03227489635).

[10] Shujun Li, Xuanqin Mou, Boliya L. Yang, Zhen Ji, and Jihong Zhang. Problems with a probabilistic encryption scheme based on chaotic systems. *International Journal of Bifurcation and Chaos*, 13(10):3063–3077, 2003. (**SCI** indexed, IDS Number: 755JY).

My Other Publication (攻读博士期间的其他文章)

[1] **Shujun Li**, Peng Wang, Xuanqin Mu, and Yuanlong Cai. Research on nonlinear dynamic systems employing color space. In *2000 5th International Conference on Signal Processing Proceedings (WCC-ICSP2000)*, volume I, pages 285–289, 2000. (**ISTP** indexed, IDS Number: BR32Z).

附件 1：

学位论文独创性声明

本人声明，所呈交的学位论文系在导师指导下本人独立完成的研究成果。文中依法引用他人的成果，均已做出明确标注或得到许可。论文内容未包含法律意义上已属于他人的任何形式的研究成果，也不包含本人已用于其他学位申请的论文或成果。

本人如违反上述声明，愿意承担以下责任何后果：

1. 交回学校授予的学位证书；

2. 学校可在相关媒体上对作者本人的行为进行通报；

3. 本人按照学校规定的方式，对因不当取得学位给学校造成的名誉损害，进行公开道歉。

4. 本人负责因论文成果不实产生的法律纠纷。

论文作者签名：_____ 日期：_____年_____月_____日

学位论文知识产权权属声明

本人在导师指导下所完成的论文及相关的职务作品，知识产权归属学校。学校享有以任何方式发表、复制、公开阅览、借阅以及申请专利等权利。本人离校后发表或者使用学位论文或与该论文直接相关的学术论文或成果时，署名单位仍然为西安交通大学。

论文作者签名：_____ 日期：_____年_____月_____日

导 师 签 名：_____ 日期：_____年_____月_____日