Кафедра «Системы Обработки Информации и Управления»«
Факультет «Информатика и Системы Управления»
Московский Государственный
Технический Университет им. Н. Э. Буамана

Institute of Simulation Sciences
Faculty of Computer Science and Engineering
De Montfort University, Leicester

ПРИЛОЖЕНИЕ ТЕОРИИ ДЕТЕРМИНИРОВАННОГО ХАОСА В КРИПТОГРАФИИ

Николай Птицын
np@beep.ru

Москва
2002

Аннотация

Настоящая работа посвящена приложению теории детерминированного хаоса (нелинейной динамики) к компьютерной криптографии. Рассмотрена взаимосвязь между хаотическими и криптографическими системами на концептуальном и практическом уровнях. Теоретическое обоснование этой связи включает обсуждение таких понятий как экспоненциальная чувствительность к начальным условиям, эргодичность, смешивание, сложность, случайность, непредсказуемость. Рассмотрены два подхода к практическому применению нелинейных системы в криптографии: (1) аппроксимация непрерывных систем при помощи математики с плавающей запятой и (2) бинарных хаос с ограниченным числом состояний. Представлен обзор публикаций с описанием хаотических шифров и хаотических псевдослучайных генераторов. Рассмотрено приложение нелинейных систем с точным решением и неоднозначным преобразованием для построения псевдослучайных генераторов.

This paper studies the application of deterministic chaos to digital cryptography. The fundamental relationship between the properties of chaotic and cryptographic systems is considered at the theoretical and practical layers. The theoretical background upon which this relationship is based, includes discussions on chaos, ergodicity, complexity, randomness, unpredictability, incompressibility. Two approaches to the finite-state implementation of chaotic systems are considered: (i) floating-point approximation of continuous-state chaos; (ii) binary pseudo-chaos. An overview is given of existing chaos-based encryption algorithms along with their strengths and weaknesses. Exactly solvable and one-step unpredictable chaotic systems are described in the context of pseudo-random generators.

Оглавление

Список иллюстраций

Вводные замечания

Трудно назвать область знания, в которой сегодня не проводились бы исследования под рубрикой, называемой синергетикой или детерминированным хаосом. Эти понятия вошли в научную картину мира сравнительно недавно, лишь в последней четверти XX века, но с тех пор интерес к ним не угасал не только в кругу математиков, физиков, химиков, биологов, но и в гуманитарных областях.

Что обычно называют синергетикой? Термин (от греч. *synergeia* — совместное действие, сотрудничество) был предложен в начале 70-х годов немецким физиком Г. Хакеном. Вначале под синергетикой понимали область научных исследований, целью которых было выявление *общих закономерностей* в процессах образования, устойчивости и разрушения упорядоченных временных и пространственных структур в сложных неравновесных системах различной природы: физических, химических, биологических, социальных и т. д. Объединяющим началом в синергетике являются объекты исследований — открытые *сложные нелинейные системы с обратными связями* [13, 15].

Понятия «детерминированность» и «хаос» в бытовом[1] или мифологическом[2] смысле на первый взгляд кажутся прямо противоположными по смыслу. Детерминизм ассоциируется с полной предсказуемостью и воспроизводимостью, хаос — с полной непредсказуемостью и невоспроизводимостью. Попробуем кратко пояснить понятие «детерминированный хаос» в его математическом смысле.

Когда говорят о детерминированности, подразумевают причинно-следственную связь. Если задано некоторое состояние системы в начальный момент времени, то оно однозначно определяет состояние системы в будущем. В детерминированной динамической системе *состояние* (т. е. совокупность внутренних независимых переменных) изменяется во времени по некоторому заданному (детерминированному) закону.

Физики разделяют динамические системы на регулярные и хаотические. Ре-

[1] Хаос — беспорядок, неразбериха [22].

[2] Гесиод, древнегреческий поэт 8–7 вв. до н. э., впервые упоминает хаос (греч. *cháos*, от *cháino* — разверзаюсь, изрыгаю) в поэме « Теогония». Хаосом называется беспредельная первобытная масса, из которой образовалось впоследствии все существующее [22].

гулярные системы являются устойчивыми, то есть малые возмущения со временем затухают, и система возвращается к исходному регулярному поведению (стационарному). Напротив, хаотическая система, неустойчива, и малые возмущения нарастают во времени. Хаотические системы обладают так называемой *экспоненциальной чувствительностью к начальным условиям*, то есть небольшое изменение начального состояние системы приводит к существенному изменению во всей траектории. Изменение в начальных условиях усиливается экспоненциально во времени. Так как реальный наблюдатель не может измерить начальные условия с абсолютной точностью, ошибка его предсказания быстро вырастает до неприемлемого уровня. Таким образом, поведение детерминированной хаотической системы может быть предсказано только вероятностно (при помощи некоторой функции плотности распределения, определяющей асимптотическую вероятность нахождения траектории в каждой области пространства состояний). Детерминированным (или динамическом) хаосом называется либо теория, изучающая хаотические системы, либо сами системы.

Сегодня математическая теория детерминированного хаоса является фундаментальной основой естествознания [17, 6, 19, 21]. Убедительно доказано, что сложность поведения хаотических систем кроется не в большом числе степеней свободы и не в наличии флуктуаций, а в экспоненциальной неустойчивости. Классическими примерами хаотического поведения является броуновское движение, изменения погоды, колебания орбит астрономических тел, поведение фондовой биржи, биологические процессы в организме человека.

В этом смысле, криптографические системы отнюдь не является исключением в многообразии окружающих нас естественных и искусственных систем и строятся по законам хаоса. Хаотические и криптографические системы взаимосвязаны на концептуальном уровне. И в криптографии, и в нелинейной динамике осуществляется *нелинейное преобразование информации*. С одной стороны, это преобразование *детерминировано* (выполняется компьютером), с другой стороны, оно должно быть *практически непредсказуемым* для внешнего наблюдателя. Таким образом, словосочетание «детерминированный хаос» вполне «подходит» для криптографии. В этой работе, предложено более формальное описание этой связи на базе синергетического подхода и теории хаоса.

Помимо концептуальной взаимосвязи, можно заметить, что и на практическом уровне криптографические и хаотические системы тоже похожи. Так, уже в начале 1950 гг. Клод Шеннон, один из основателей теории статистической связи и современной криптографии, в явном виде упомянул растягивающий и сжимающий механизм хаоса применительно к шифрованию данных: «*Good mixing transformations are often formed by repeated products of two simple noncommuting*

operations. Hopf[3]*, has shown, for example, that pastry dough can be mixed by such a sequence of operations. The dough is first rolled out into a thin slab, then folded over, then rolled, and then folded again, etc...»*[4][79] Фундаментальное криптографическое свойство сжимающего и растягивающего преобразования будет подробнее рассмотрено в этой работе.

Однако, глубокие исследования этой области стали проводиться значительно позднее, вместе с формированием современной теории хаоса и развитием вычислительной техники. Наступление «информационного века» и осознание созидательной и разрушительной силы знаний стали причиной особой актуальности криптографии сегодня. Помимо использования отработанных схем защиты информации, наблюдается непрерывный поиск новых технологий. В большей степени, это обусловлено не столько желанием увеличивать *количественно* криптостойкость традиционных схем шифрования (имеющих и без того большой запас прочности), а, сколько, необходимостью не зависеть от существующих стандартов и «нерешенных» математических проблем, которые, внезапно могут перестать быть препятствием перед злоумышленником.

Новые открытия зачастую возникают на стыке нескольких научных областей, когда теоретическая модель, изученная в одной предметной области, метафорически переносится в другую и начинает работать. Так, другой целью этой работы является поиск новых приложений теории детерминированного хаоса в криптографии. Немало ученых уже пробовало применить модели из физики, биологии для шифрования информации. Мы попробуем структурировать и обобщить этот опыт, а так же показать, что все традиционные криптографические системы могут рассматриваться в рамках синергетического подхода (то есть как нелинейные динамические системы).

[3] Хопф, Э.Ф.Ф. (1902–1983), австрийский математик, сделал существенный вклад в теорию эргодичности и топологии.

[4] Хорошее смешивающее преобразование часто получают при помощи многократного применения двух простых некоммутирующих операций. Хопф например, показал, что тесто может быть смешано при помощи последовательности таких операций. Сначала тесто скатывается в тонкий длинный кусок, потом складывается пополам, потом снова скатывается в тонкий кусок и складывается, и так далее...

Глава 1

Основные принципы и определения

В главе рассматривается взаимосвязь между криптографическими системами и динамическими системами в теории детерминированного хаоса. Представлены базовые понятия из криптографии (Б. Шнайера [77], А. Менезеса [69]) и нелинейной динамики (Дж. Д. Биркгоф [11], П. Р. Халмош [12], Я. Г. Синай [9]).

1.1 Криптография

Криптография занимается проблемой защиты информации путем ее преобразования. Криптография решает задачи конфиденциальности, аутентификации, целостности и ряд других с ними связанных. Практическая криптография изучает методы шифрования данных, управления ключами и сертификатами, создания цифровой подписи. Криптоанализ решает условно противоположенные задачи, в частности, преодоление защиты и несанкционированное дешифрование данных (без знания ключа). Криптология — это раздел математики, изучающий математические основы методов криптографии и криптоанализа.

1.1.1 Криптографическая система

В широком смысле, криптографическая система есть вся инфраструктура, обеспечивающая защиту информации (средствами вычислительными техники), то есть совокупность согласованных средств шифрования, передачи ключей, аутентификации и других компонент. В реальной жизни, криптосистема представляет собой аппаратно-программный комплекс, взаимодействующий с человеком.

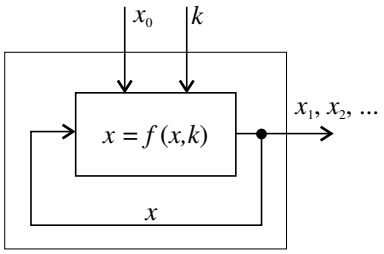

Рис. 1.1. Криптосистема

В узком математическом смысле, криптосистема $\mathcal{S} = \langle X, Y, \mathcal{K}, f \rangle$ есть некоторые преобразование информации $f : X \times \mathcal{K} \to Y$, определенное на множествах исходных состояний X, заключительных состояний Y и ключей $\mathcal{K}$. Состояние $x \in X$ кодирует некоторую полезную информацию. В компьютерной криптографии множества $X = Y = \subset \{0,1\}^*, \mathcal{K} \subset \{0,1\}^*$, а преобразование f задано при помощи программы (алгоритма), реализуемого на машине Тьюринга (раздел 2.2.1).

Преобразование f может рассматриваться в качестве итерации криптографического алгоритма (рис. 1.1). Тогда криптосистема производит последовательность состояний $x_0, x_1, \ldots, x_i, \ldots$, где $x_i = f(x_{i-1}, k) = f^i(x_0, k)$, $x_0 \in X$, $k \in \mathcal{K}$. Это последовательность называется траекторией или орбитой системы. Вся траектория определяется начальным состоянием системы x_0 и параметром k.

Последовательное преобразование состояний системы в результате применения некоторой однотипной элементарной функции f можно наблюдать в блочных и поточных шифрах, генераторах псевдослучайных чисел, односторонних функциях. Такие системы являются компонентам криптосистемы в широком смысле.

Таким образом, под криптосистемой в узком смысле можно понимать динамические систему $\langle f, X, \mathcal{K} \rangle$ с нелинейной функцией f, пространством состояний X и пространством параметром K. Как будет показано ниже, требования к криптосистемам оказываются взаимосвязаны с такими свойствами динамических систем как хаотичность, эргодичность и смешивание.

1.1.2 Шифрование и дешифрование

В качестве информации, подлежащей защите, будем рассматривать текстовые сообщения p. Сообщение, называемое так же открытом текстом (plaintext), есть

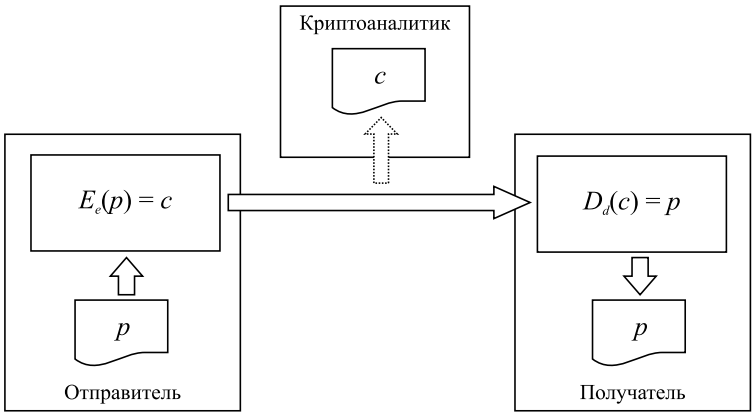

Рис. 1.2. Схема защищенной передачи сообщения при помощи шифрования и дешифрования.

последовательность символов

$$p = \{p_1, p_2, \ldots, p_n \,|\, p_i \in \mathcal{P}\},$$

где алфавит $\mathcal{P}$ есть конечное множество символов, используемых для кодирования информации. В компьютерных криптосистемах, $\mathcal{P} = \mathbb{Z}_2 = \{0, 1\}$ (бинарный алфавит). Можно рассматривать символы на алфавите байтов, т. е. $\mathcal{P} = \mathbb{Z}_{256}$.

Операция *шифрование* есть криптографическое преобразование $E : \mathcal{P}^* \times \mathcal{E} \to \mathcal{C}^*$, где $\mathcal{C}$ — алфавит шифротекста, и $\mathcal{E}$ — множество ключей шифрования, то есть

$$c = E(p, e) = E_e(p), \quad p \in \mathcal{P}, c \in \mathcal{C}, e \in \mathcal{E}.$$

Операция *дешифрование* есть обратное преобразование $D : \mathcal{C}^* \times \mathcal{D} \to \mathcal{P}^*$, где D — множество ключей дешифрования, то есть

$$p = D(c, d) = D_d(c), \quad p \in \mathcal{P}, c \in \mathcal{C}, d \in \mathcal{D}.$$

Обычно, $\mathcal{C} = \mathcal{P} \subset \{0, 1\}^*$ и $\mathcal{E} = \mathcal{D} = \mathcal{K} \subset \{0, 1\}^*$.

На рис. 1.2 представлена классическая схема защищенной передачи сообщения при помощи шифрования/дешифрования.

1.1.3 Схема шифрования

Определение 1. Схема шифрования (кратко, шифр) есть структура вида

$$\mathcal{S} = \langle E, D, \mathcal{P}, \mathcal{C}, \mathcal{K} \rangle,$$

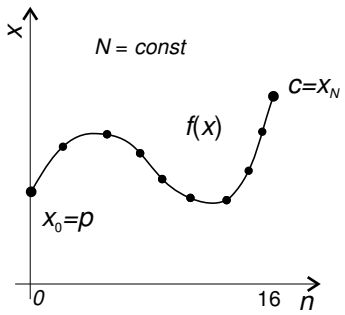

Рис. 1.3. Траектория блочной криптосистемы. Каждый блок шифруется при помощи *отдельной* траектории, причем начальным состоянием является открытый текст, а заключительным — шифротекст. Система осуществляет фиксированное число итераций N. Последующий блок шифруется на базе другой траектории.

где $E : \mathcal{P}^* \times \mathcal{K} \to \mathcal{C}^*$ и $D : \mathcal{C}^* \times \mathcal{K} \to \mathcal{P}^*$, таких что для каждого ключа $e \in \mathcal{K}$ существует уникальный ключ $d \in \mathcal{K}$ и $D_d = E_e^{-1}$, то есть

$$\forall p \in \mathcal{P}, e \in \mathcal{K}, \quad \exists d \in \mathcal{K} : \quad p = D\big(E\left(p, e\right), d\big).$$

Практически, схема задается алгоритмами E, D и множествами $\mathcal{P}, \mathcal{C}$ и $\mathcal{K}$.

Секретность некоторых схем шифрования основана на том, что сам алгоритм шифрования (дешифрования) является секретным, то есть неизвестными криптоаналитику. Сегодня такие схемы представляют лишь исторический интерес и не имеют практического значения. В современных алгоритмах секретность преобразования целиком заложена в ключе (принцип Кирхгофа).

Симметричные и асимметричные схемы

Различают симметричные и асимметричные схемы шифрования. В *симметричных схемах* (или схемах с секретным ключом) используют один и тот же ключ для шифрования и дешифрования, т. е. $p = D\left(E\left(p, k\right), k\right)$, $k = e = d$ (или же ключ d просто вычисляется из e). В *асимметричных схемах* (или схемах с открытым ключом) ключи e и d не совпадают, и из одного (так называемого открытого) ключа практически невозможно получить парный ему ключ.

Блочные и потоковые схемы

Другая известная классификация выделяет блочные и потоковые шифры. Особенность *блочной схемы* заключается в том, что шифрование осуществляется

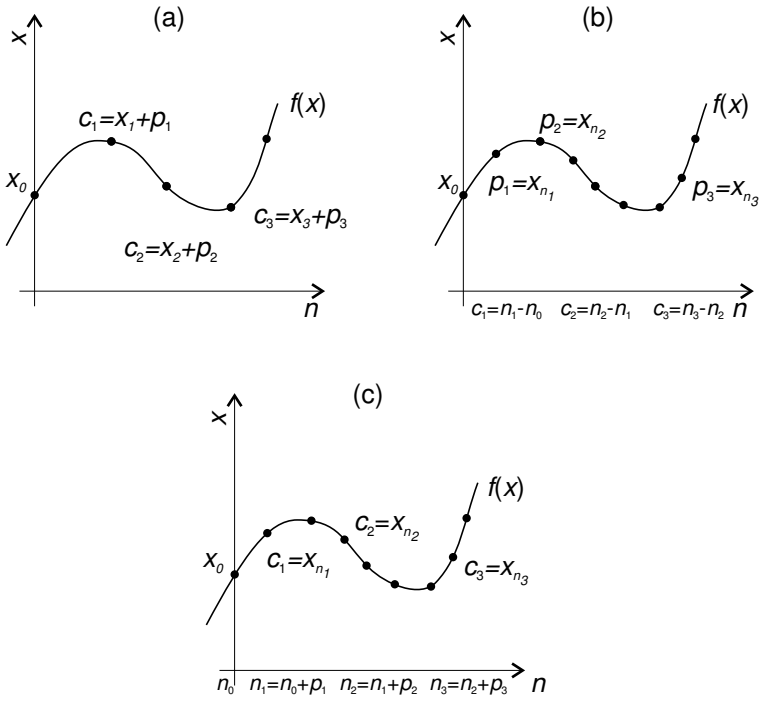

Рис. 1.4. Траектории потоковой криптосистемы. (a) Шифротекст c есть простая сумма открытого текста p и текущего состояния x (Бианко, 1991); (b) Шифротекст c есть число итераций n (Вонг, 1999; Баптиста, 2000); (c) Шифротекст c есть заключительное состояние системы после p итераций (Гелафер, 1996).

блокам. Каждый блок шифруется (дешифруется) независимо от других. Идентичные блоки открытого текста будут преобразованы в идентичные блоки шифротекста. С другой стороны, *поточная схема* (так же называемая шифром состояния) преобразует поток символов (блоков) текста в поток шифротекста, причем преобразование зависит от состояния системы. Идентичные символы (блоки) текста будут вероятнее зашифрованы в различные символы (блоки) шифротекста.

Как уже отмечалось, схемы шифрования можно рассматривать с точки зрения нелинейной динамики. Остановимся подробнее на симметричных схемах.

1) Шифрование блочного алгоритма осуществляется путем n-кратного применения некоторой итерационной функции f (рис. 1.3). Число n фиксировано и невелико (часто, $n = 16$). Каждая итерация переводит криптосистему в следующее состояние, то есть $x_{i+1} = f(x_i)$. Начальному состоянию присваивается открытый текст ($x_0 = p$), а заключительное состояние принимается за шифротекст ($c = x_n$).
2) Потоковые схемы более разновидны с точки зрения использования траектории (рис. 1.4). Их отличительной чертой является то, что весь шифротекст «связан» одной траекторией, то есть шифрование порции открытого текста зависит от текущего состояния криптосистемы. Число итераций не фиксировано, а зависит от объема исходного текста.

Примеры блочных и поточных криптосистем, использующие хаос в явном виде, будут рассмотрены в Главе 3.

Шифр Вернама

Простейшей потоковой схемой является шифр Вернама [69]. Шифрование осуществляется путем побитного сложения по модулю 2 открытого текста и ключевой последовательности.

$$c_i = p_i \oplus k_i$$

Дешифрование получается повторным прибавлением ключа:

$$p_i = c_i \oplus k_i,$$

так как $p_i \oplus k_i \oplus k_i = p_i$.

На рис. 1.4 (а) представлена аналогичная схема шифрования. Траектория криптографической системы f соответствует ключевой последовательности. После каждой итерации происходит шифрование $c_i = p_i + x_i$ (дешифрование $p_i = c_i - x_i$).

Если ключевая последовательности $k = \{k_i\}$ является истинно случайной, то шифр Вернама называется одноразовым блокнотом. Подробно понятия случайности и идеальной безопасности будут рассмотрены в Главе 2.

1.1.4 Псевдослучайный генератор

Клод Шеннон [78] показал, что симметричная схема шифрования (например, шифр Вернама) безусловно безопасна только в том случае, если ключевая последовательность k имеет равномерный закон распределения (истинно случайна) и ее битовая длина равна длине исходного сообщения p (например, в случае одноразового блокнота). На практике такие ключи генерировать и передавать весьма затруднительно. Вместо них используют так называемые псевдослучайные последовательности. Псевдослучайные последовательности «кажутся» случайными, то есть их закон распределения не может быть эффективно отличен от равномерного закона (доступными вычислительными средствами). С другой стороны, псевдослучайная последовательность порождается некоторым детерминированным генератором из короткого ключа (семени), она легко репродуцируема. Грубо говоря, случайность (неопределенность) семени «размазывается» по всей последовательности.

Псевдослучайные генераторы играют центральную роль в современной криптографии. Так как алгоритм криптографического преобразования должен оставаться постоянным в течении шифрования всего сообщения, меняются только его параметры. Числа псевдослучайной последовательности используются в качестве параметров криптографического преобразования. Таким образом псевдослучайный генератор задает всю цепочку криптографических преобразований.

Согласно выбранному подходу, мы будем рассматривать псевдослучайный генератор как динамическую систему. На рис. 1.1 представлена система, порождающая последовательность чисел. Любая последовательность, которая может быть сгенерирована системой, задается начальными состоянием x_0 и параметром k.

Важным требованием к динамической системе, используемой для генерации ключевых последовательностей, является псевдослучайность и непредсказуемость. В следующих разделах мы рассмотрим эти понятия с точки зрения теории хаоса и нелинейной динамики. Формальное определения псевдослучайного генератора представлено в Главе 2.

1.1.5 Запутывание и распыление

Истинно случайная ключевая последовательность позволяет полностью устранить статистические инвариантности криптографического преобразования. Однако, как уже было отмечено, в криптографии используют псевдослучайные последовательности, поэтому некоторая часть информации об исходном тексте «просачивается» в шифротекст. Так как исходный текст, как правило, обладает известной избыточностью, криптоанализ становятся теоретически возможными

уже при наличии информации о статистических свойствах сообщения. Избыточность сообщения может быть понижена при помощи хорошей компрессии. Несжимаемое сообщение характеризуются тем, что изменение хотя бы одного его бита, приводит к полному изменению его смысла.

Если сообщение не может быть сжато до теоретического минимума, то, согласно Шеннону, необходимо использовать две базовые техники для скрытия избыточности:

Запутывание (confusion) предполагает, что статистические свойства открытого текста не заметны в шифротексте. Для всякого наблюдателя шифротекст должен казаться случайным, то есть быть псевдослучайным.

Распыление (diffusion)

1) в смысле текста, распыление предполагает, что статистически похожие последовательности текста преобразуются в совершенно различные последовательности шифротекста (при шифровании одним и тем же ключом). Другими словами, любой элемент открытого текста должен влиять на все элементы шифротекста сложным непредсказуемом образом. Так, изменение одного бита сообщения должно приводить к изменению половины битов шифротекста.
2) в смысле ключа, распыление предполагает, что похожие ключи преобразуют текст в совершенно разные шифротексты. Аналогично, каждый бит ключа должен влиять на каждый бит шифротекста сложным непредсказуемым образом. Это же свойство должно иметь место и при дешифровании, так как в противном случае криптоаналитик имеет возможность определить факт того, что часть ключа угадана верно.

Этот подход в явном виде реализован в симметричных блочных шифрах. Итерационная функция обычного блочного алгоритма включает фазы подстановки (замены, substitution) и перемешивания (permutation). Так в классическом алгоритме DES фазы реализованы при помощи таблиц подстановки (s-box) и перемешивания (p-box). Считается, что фаза перестановки обеспечивает запутывание, а фаза перемешивания — распыление. В конечном счете, оба свойства обеспечивают псевдослучайность шифротекста для любого текста и любого ключа. Перемешивания является эффективным средством усиления нелинейности итерационной функции криптосистемы.

1.2 Хаос и криптография

1.2.1 Динамическая система

Динамическая система непрерывного состояния и непрерывного времени $\mathcal{S} = \langle X, \mathcal{K}, f \rangle$, зависящая от параметров, может быть задана дифференциальным уравнением

$$\frac{dx}{dt} = f(x, k), \quad x \in X \subseteq R^d, k \in \mathcal{K} \subseteq R^{d_K},$$

где $f : X \times \mathcal{K} \to Y$ — гладкая вектор-функция, X — пространство состояний и $\mathcal{K}$ — пространство управляющий параметров. Для каждого начального условия x_0 система удовлетворяет условию существования и единственности решения $x(t, x_0)$, где $x(0, x_0) = x_0$ [11]. Кривая $\varphi_t(t, x_0)$, соответствующая этому решению, называется траекторией.

Динамическая система непрерывного состояния (*дискретного* времени) может быть задана итерационной функцией

$$x_{n+1} = f(x_n, k), \quad x_n \in X \subseteq R^d, k \in \mathcal{K} \subseteq R^{d_K}, n = 0, 1, 2, \ldots \tag{1.1}$$

где x_i — дискретные состояния системы. Траектория $\varphi(i, x_0)$ представляет собой последовательность $x_0, x_1, x_2, \ldots$ Легко заменить, что выражение (1.1) похоже на криптографическую итерационную функцию, используемую в псевдослучайных генераторах, блочных шифрах и пр. (см. рис. 1.1–1.4): в динамической и в криптографической системах мы имеем дело с *итерационном преобразованием информации, зависящим от параметра*.

Ниже мы будет опускать параметр k в обозначениях системы $\langle X, f \rangle$ и итерационной функции $f(x)$. Результат n-кратного применения $f(x)$ будем записывать в виде

$$x_n = f((\cdots f(x_0) \cdots)) = f^n(x_0), \quad x_0, x_n \in X.$$

1.2.2 Хаотическая система

Исследователями отмечено несколько признаков при которых наблюдается хаотическое поведение системы [30]. В частности, необходимым условием являются два классических свойства — топологическая транзитивность и чувствительность к начальным условиям.

Определение 2 (хаотическая система). Динамическая система $\langle X, f \rangle$ называется хаотической, если выполняются условия:

1) Функция $f : X \to X$ *топологически транзитивна* на некотором метрическом множестве $X \subset \mathbb{R}^d$, если для любых открытых множеств $U, V \subset X$

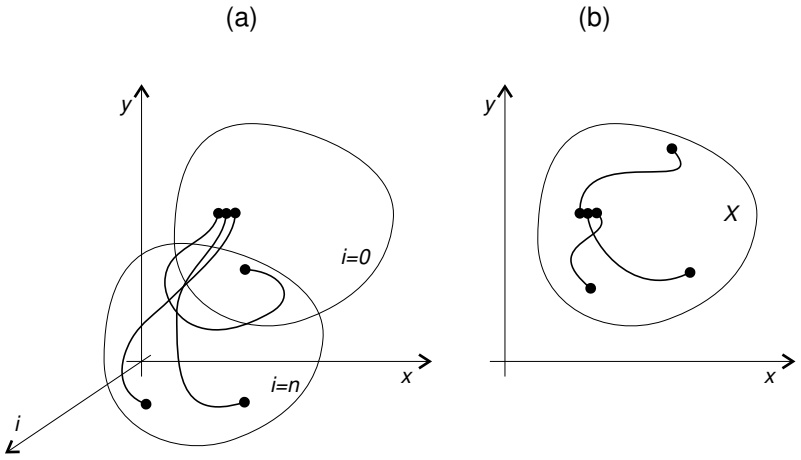

Рис. 1.5. Двумерная хаотическая система: (a) Временное пространство; (b) Фазовое пространство.

существует $n \geq 0$, такое что

$$f^n(U) \cap V \neq \emptyset.$$

2) Функция f *чувствительна к начальным условиям*, если существует $\delta > 0, n \geq 0$, такое что для любого $x \in X$ и его окрестности H_x есть $y \in H_x$ для которого

$$|f^n(x) - f^n(y)| > \delta.$$

Другими словами, динамическая системы называется хаотической, если все ее траектории ограничены, но быстро расходятся в каждой точке фазового пространства (рис. 1.5)[1].

Криптосистемы, рассмотренные в разделе 1.1, по своим требованиям похожи на хаотические системы: топологическая транзитивность необходима, с одной стороны, для сохранения состояния криптосистемы в тех пределах, которые допускает носитель информации, а, с другой стороны, для «покрытия» всего пространства состояний шифротекста. Чувствительность к начальным условиям соответствует чувствительности криптосистемы к открытом тексту или семени

[1]На рис. 1.5 условно показана *двухмерная* система с *непрерывной* траекторией. Ясно, что двумерная система не может быть непрерывной и хаотической одновременно. В противном случае любое пересечение траекторий приводило бы к образованию циклических орбит. Напротив, непрерывные системы с размерностью три и более могут быть хаотическими [10]. Это замечание не относится к системам дискретного времени, задаваемые выражением вида (1.1), в которых траектория представляет собой последовательность дискретных значений.

псевдослучайного генератора (см. *распыление* в разделе 1.1.5). Таким образом, *и в теории хаоса, и в криптографии мы имеем дело с системами, в которых небольшое изменение начальных условий приводит к существенным изменениям во всей траектории.*

1.2.3 Экспоненты Ляпунова

В определении хаотической системы (2) мы ввели понятие чувствительности к начальным условиям. Показатель Ляпунова $\lambda(x_0)$, определяемый для каждой точки $x_0 \in X$, является мерой чувствительности, то есть характеризует скорость экспоненциального разбегания траекторий, находящихся в окрестности x_0 [17]. Для одномерной системы

$$|f^n(x_0 + \varepsilon) - f^n(x_0)| = \varepsilon \cdot e^{n\lambda(x_0)},$$

где ε — небольшое отклонение от начального состояния x_0, и n — число итераций (дискретное время). В общем случае, λ зависит от начальных условий x_0, поэтому определяют усредненное значение. Для систем, сохраняющих меру (1.2.5), λ остается постоянным для всех траекторий. Практически, показатель Ляпунова можно вычислить как предел

$$\lambda(x_0) = \lim_{n\to\infty} \lim_{\varepsilon\to 0} \frac{1}{n} \log \left| \frac{f^n(x_0 + \varepsilon) - f^n(x_0)}{\varepsilon} \right| \tag{1.2}$$

или

$$\lambda(x_0) = \lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \log \left| f'(x_k) \right| = \lim_{n\to\infty} \frac{1}{n} \log \prod_{k=1}^{n} \left| f'(x_k) \right| \tag{1.3}$$

Для каждого k, производная $f'(x_k)$ показывает как быстро изменяется функция f по отношению к возрастанию аргумента с x_k до x_{k+1}. Предел равен среднему значению логарифма производной после n итераций и показывает скорость расхождения траекторий в течении дискретного времени n. Положительное значение показателя ($\lambda > 0$) есть индикатор хаотического поведения системы.

Для d-мерной системы мы имеем набор $\lambda = \{\lambda_1, \ldots, \lambda_d\}$ и более сложное поведение, которое качественно не отличается от одномерного случая [70].

Для учета разрешения (точности) наблюдения, более полезной информацией оказывается энтропия Колмогорова-Синая h_{KS}, которая будет рассмотрена в Главе 2.

С позиции криптографии, показатель Ляпунова является мерой криптографической эффективности системы. Чем больше λ, тем меньше итераций требуется для достижения заданной степени распыления или смешивания информации.

1.2.4 Бифуркация

Бифуркацией (от лат. *bifurcus* - раздвоенный) обычно называют процесс качественного перехода от регулярного поведения к хаотическому в результате изменения управляющего параметра. Например, удвоение Фейгенбаума является одним из типов бифуркаций (рис. 3.9). В точках бифуркаций происходит удвоение числа устойчивых состояний (периодов). С увеличением параметра удвоение происходит все чаще и чаще, и приводит к хаотическому режиму системы.

В криптографических приложениях выбор значения управляющего параметра определяет непредсказуемость системы. Если параметр используется в качестве ключа, то все пространство допустимых ключей должно соответствовать хаотическому режиму.

1.2.5 Эргодические и смешивающие системы

Пусть динамическая система $\mathcal{S} = \langle X, f \rangle$ имеет f-инвариантную меру μ, $\mu(X) < \infty$, то есть

$$\forall A \in \sigma(X), \qquad \mu(A) = \mu\left(f^{-1}(A)\right),$$

где $\sigma(X)$ есть σ-алгебра измеримых подмножеств X. Пусть f-инвариантная мера эквивалента мере Лебега с функцией плотности распределения $g(x)$, ограниченной некоторыми положительными константами g_1 и g_2:

$$0 < g_1 < g(x) < g_2,$$

где $\forall A \in \sigma(X), \quad \mu(A) = \int_A g(x)\,dx$. Если g_1 близко к g_2, то мера μ близка к равномерному закону распределения.

Динамическая система называется *эргодической*, если существуют только тривиальные инвариантные множества, то есть, для любого измеримого множества A, f-инвариантного относительно меры μ, имеем либо $\mu(A) = 0$, либо $\mu(X \setminus A) = 0$ [12]. Эргодичность подразумевает, что пространство X не может быть разделено на f-инвариантные нетривиальные и непересекающиеся подмножества. С точки зрения криптографии, это свойство обеспечивает максимальную устойчивость против криптоанализа методом перебора (brute-force — атаки грубой силой), так как криптоаналитик должен вести поиск по всему пространству состояний X и не может ограничиться некоторым «предполагаемым» подмножеством. Свойство смешивания тесно связано с шенноновским распылением информации.

Динамическая система называется *смешивающей*, если выполнено условие

$$\forall C, P \in \sigma(X), \qquad \lim_{n\to\infty}\left(\frac{\mu\left(f^{-n}(C) \cap P\right)}{\mu(P)}\right) = \frac{\mu(C)}{\mu(X)}.$$

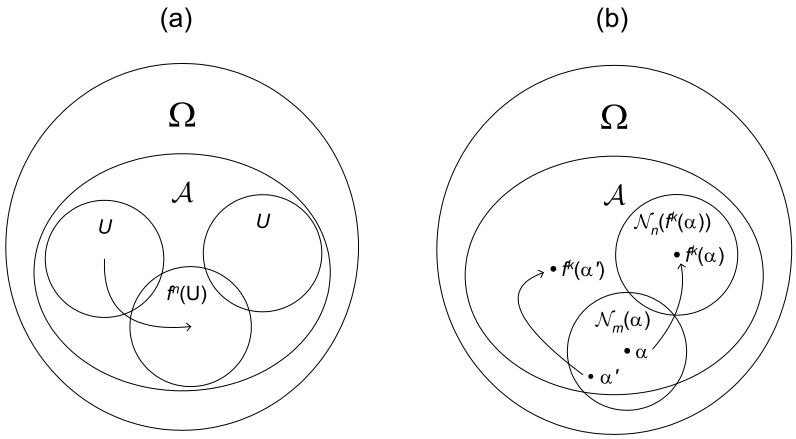

Рис. 1.6. Дискретный хаос: свойства чувствительности к начальным условиям (a) и топологической транзитивности (b) определены на множестве Ω.

Если $\mu(X) = 1$ (мера μ вероятностная), то

$$\lim_{n\to\infty} \left(\mu\left(f^{-n}(C) \cap P\right)\right) = \mu(C)\,\mu(P).$$

Смешивающие свойство подразумевает, что та часть P, которая попадает в C после n преобразований f, ассимтотически пропорциональна размеру C в X (в смысле меры μ). Более того, многократные преобразование f делает любое множество C статистически независимым от P (асимптотически). Другими словами, траектория, с началом в $x_0 \in X$ после $n \to \infty$ итераций будет появляться в окрестностях каждой точки с одной и той же вероятностью. И наоборот, для фиксированного конечного состояния x_n и достаточного большого n, начальное состояние x_0 будет μ-равновероятно [61]. Таким образом, x_0 и x_n оказываются асимптотически независимыми (см. Главу 2).

1.2.6 Бинарный хаос

Мы дали определения и рассмотрели некоторые свойства хаотических систем с d-мерном пространством состояний, определенным на множестве действительных чисел. Однако, компьютерная криптография построена на дискретных системах, в которых состояние системы определяется бинарной (двоичной) последовательностью. Классическими примерами бинарных систем являются клеточные автоматы или дискретные нейросети.

В 1998 Ваелброек (Waelbroeck) и Зертуче (Zertuche) [83] предложили теорию

дискретного хаоса для бинарных систем. Рассмотрим пространство состояний

$$\Omega = \{\alpha | \alpha \in \mathbb{Z}_2^*\}, \quad \mathbb{Z}_2 = \{0, 1\}.$$

Состояние системы $\alpha \in \Omega$ задано последовательностью символов (битов)

$$\alpha = \alpha(1)\alpha(2)\ldots\alpha(i)\ldots\alpha(n).$$

Естественной мерой для бинарных систем является Хеммингово расстояние

$$d_H(\alpha, \alpha') \equiv \sum_{i=1}^{n} \left|\alpha(i) - \alpha'(i)\right|,$$

то есть число бит, которые не совпадают в строках α и α'. Однако, в пределе $n \to \infty$ (для бесконечных строк) Хеммингово расстояние не может задавать топологию пространства Ω. Для предельного случая удобно ввести базу

$$\mathcal{N}_n(\alpha) = \left\{\alpha' \in \Omega | \alpha(i) = \alpha'(i), \forall i < n\right\},$$

где $n = 1, 2, 3\ldots$. Тогда, определение хаоса можно обобщить с R^d на Ω:

Определение 3 (бинарный хаос, [83]). Пусть задано компактное множество $\mathcal{A} \subset \Omega$ и отображение $f : \mathcal{A} \to \mathcal{A}$. Бинарная система $\langle \mathcal{A}, f \rangle$ называется хаотической, если выполнены два условия:

1) Функция f является *топологически транзитивной* на множестве $\mathcal{A}$, то есть
   для любых открытых подмножеств $U, V \subset \mathcal{A}$ существует $n \in \mathbb{Z}$ такое, что $f^n(U) \cap V \neq \emptyset$ (рис. 1.6-b)[2].
2) Функция f является *чувствительной к начальным условиям* на множестве $\mathcal{A}$, то есть для любой строки $\alpha \in \mathcal{A}$ и множества $\mathcal{N}_m(\alpha)$ существуют $n, k \in \mathbb{N}$ и $\alpha' \in \big(\mathcal{N}_m(\alpha) \cap \mathcal{A}\big)$, такие что $f^k(\alpha') \notin \mathcal{N}_n\big(f^k(\alpha)\big)$ (рис. 1.6-a).

Таким образом, для дискретных бинарных систем хаотическая система может быть определена по аналогии с непрерывными. Так же можно ввести понятия показателя Ляпунова, хаотического аттрактора, эргодичности, смешивание и т. д. В Главе 3 мы уведем на примерах, что классические криптосистемы могут рассматриваться как бинарного хаоса с ограниченным числом состояний.

1.3 Взаимосвязь

Мы рассмотрели некоторые понятия из криптографии и хаотической динамики. Традиционные криптосистемы (схемы шифрования, псевдослучайные генераторы) можно рассматривать как динамические системы, осуществляющие преобразования информации (табл. 1.1).

[2] Если $n < 0$ и функция f не имеет обратной, то считаем $f^n(U) = \{\alpha \in \mathcal{A} | f^{-n}(\alpha) \in U\}$.

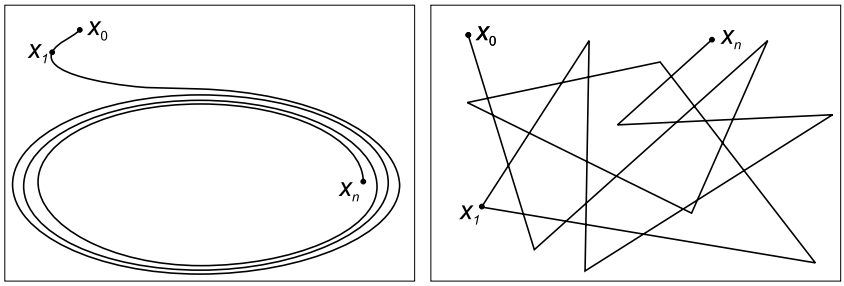

Рис. 1.7. Пример фазовых портретов хаотической и криптографической систем. Хаотическая система может иметь дробную размерность меньшую, чем число независимых переменных системы (слева). В криптографических системах стараются использовать все пространство с максимальной, целой размерностью (справа).

Таблица 1.1. Взаимосвязь между объектами изучения в теории хаоса и криптографии

Теория хаоса	**Криптография**
хаотическая система	псевдохаотическая система
— нелинейное преобразование	— нелинейное преобразование
— бесконечное число состояний	— конечное число состояний
— бесконечное число итераций	— конечное число итераций
начальное состояние	открытый текст
заключительное состояние	шифротекст
начальные условия и параметры	ключ
асимптотическая независимость начального и конечного состояний	запутывание
чувствительность к начальным условиям и параметрам, смешивание	распыление

Можно предположить, что известные свойства хаотических систем (экспоненциальное расхождение траекторий, эргодичность, смешивание) окажутся полезными в криптографии (в частности, при разработке новых схем шифрования). В Главе 2, мы подробно рассмотрим эти свойства, используя аппарат теории сложности и теории вероятности.

С точки зрения акцентов и объектов изучения, между криптографией и теорией хаоса существуют фундаментальные различия:

1) Криптография изучает *эффект конечного числа итерационных преобразований* ($n < \infty$), в то время как теория хаоса (непрерывного и дискретного) изучает *асимптотическое поведение* системы ($n \to \infty$).
2) Классические хаотические системы представлены некоторым объектом (множеством) фазового пространства, который часто имеет дробную размерность (то есть является фракталом). В криптографии, мы пытаемся использовать все возможные комбинации независимых переменных (что делает систему максимально непредсказуемой) и работаем с пространствами с целыми размерностями (рис. 1.7).
3) Важно, что в компьютерной криптографии рассматриваются системы с *конечном числом состояний*, а пространство состояний хаотической системы определено на *бесконечном множестве* непрерывных или дискретных значений. Таким образом, все модели хаоса, реализованные на компьютере являются приближенными. Поведение таких аппроксимаций хаоса (псевдохаоса) и их приложение в криптографии будет рассмотрено в Главе 3.

Глава 2

Случайность, сложность, хаос

Понятия случайности, непредсказуемости, сложности и энтропии являются концептуальной основой криптографии. Можно сказать, что создание любой криптосистемы (например, схем шифрования или обмена ключей) сводится к разработке такого преобразования, которое было бы в нужной степени *непредсказуем* для внешнего наблюдателя.

Настоящая глава обобщает и связывает эти понятия со свойствами хаотической системы. Главной целью является построение генератора криптографических последовательностей на базе хаотической и смешивающей системы.

2.1 Неформальный обзор

Под криптографическим объектом будем понимать некоторый объект криптоанализа — криптосистему (псевдослучайный генератор, шифр) или некоторую последовательность ее состояний (реализацию). Наблюдая за объектом, криптоаналитик может охарактеризовать его как «непредсказуемый», «случайный», «сложный». Что скрывается за этими свойствами? Каким образом они связаны с хаотической системой?

Идеальная безопасность (perfect security) объекта имеет место только в том случае, если он **абсолютно непредсказуем** для внешнего наблюдателя (криптоаналитика). Это подразумевает, что все возможные исходы (состояния) равновероятны и не зависят от предыдущих состояний. Другими словами, последовательность состояний характеризуется равномерным законом распределения вероятности и не имеет корреляций (паттернов). Понятие абсолютной непредсказуемости эквивалентно **истинной случайности**. Так же, истинно случайная последовательность часто называется **белым шумом**. Источником белого шума может быть хаотическая система, с большим количеством степеней свободы

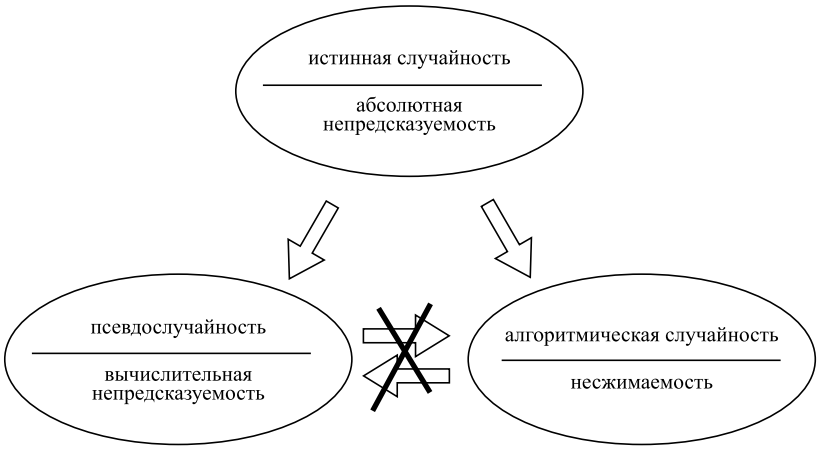

Рис. 2.1. Истинная случайность, псевдослучайность, алгоритмическая случайность и эквивалентные классы понятий.

(например, замкнутая система с идеальным газом).

В реальном мире, криптографические системы обеспечивают некоторую **практическую безопасность**, которая в существенной степени меньше, чем идеальная (в силу эксплуатационной и экономической целесообразности). Понятия случайности и непредсказуемости заменяются соответственно на псевдослучайность и вычислительную (полиномиальную) непредсказуемость. **Псевдослучайный** объект не может быть отличен от истинно случайного объекта при помощи доступных вычислительных средств внешнего наблюдателя. Аналогично, поведение **вычислительно непредсказуемого** объекта не может быть предсказано вычислительными средствами наблюдателя. Можно доказать, что псевдослучайный объект, является вычислительно непредсказуемым.

Наряду с вероятностными свойства объекта мы будем рассматривать алгоритмическую сложность. Внутренняя сложность обуславливает внешнюю непредсказуемость. Объект называется **алгоритмически случайным**, если размер самой короткой компьютерной программы (число элементарный команд), которая его порождает, не меньше размера объекта (длины последовательности). По определению, алгоритмически случайный объект **несжимаем**, так не содержит заметной избыточности и паттернов.

Конечно, истинно случайный объект является алгоритмически случайным и псевдослучайным (рис. 2.1). Однако, понятия псевдослучайности и алгоритмической случайности различаются: псевдослучайная строка создается компактным генератором, но внешний наблюдатель не может построить этот генератор

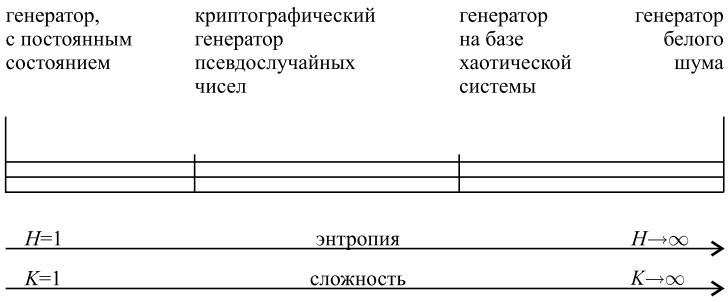

Рис. 2.2. Шкала непредсказуемости

и предсказать последовательность. Другими словами, для криптографа псевдослучайная последовательность сильно сжимаема, а для криптоаналитика — вычислительно несжимаема. С другой стороны, совсем необязательно, что алгоритмически случайная последовательность псевдослучайна или вычислительно непредсказуема. Алгоритмически случайный объект может быть предсказан при помощи вероятностной машины.

Противоположностью абсолютной непредсказуемости, является «абсолютная предсказуемость», то есть когда криптоаналитик владеет полной и достоверной информацией об объекте. Между этими двумя крайностями существует бесчисленное множество промежуточных состояний, которым соответствуют, например, системы в биологии, физики или криптографии. Количественными координатами на шкале рис. 2.2 (то есть мерой непредсказуемости или случайности) могут служить такие величины как энтропия (теоретико-информационный подход) и алгоритмическая сложность (теоретико-сложностный подход). Шенноновская энтропия H определяет степень нашего незнания о системе. Энтропия определяется числом возможных исходов (числом наблюдаемых последовательностей состояний) и функцией вероятности их наступления. Максимальное значение этой величины при постоянном числе возможных исходов достигается когда все исходы равновероятны. Сложность K, грубо говоря, есть длина минимальной программы, задающей поведение системы. Интуитивно, системы, характеризуемые большими значениями энтропии и сложности, будут «сильно непредсказуемы». При некоторых предположениях можно показать, что энтропия и сложность количественно связаны.

Какое место на этой шкале (рис. 2.2) занимает детерминированных хаос?

Естественный хаос (вещество, природа, вселенная) обладает колоссальной размерностью, бесчисленном множеством состояний и неохватимой сложностью «системы итерационных функций». Тем не менее, за счет самоорганизации эн-

тропия таких систем существенно меньше, чем у «совсем случайной» системы соответствующего масштаба. Многомерные хаотические системы не могут использоваться в шифровании, так как они не репродуцируемы. С другой стороны, генерация ключей (без возможности повтора) при помощи «естественного» хаоса (например, термальный шум в системно блоке компьютера) широко используется уже сегодня.

Детерминированный хаос, о котором уже шла речь в разделе 1.2.2 и который мы собираемся применить в шифровании, имеет малую размерность и бесчисленное множество состояний. Очевидно, такие системы «более предсказуемы», чем естественных хаос, но могут моделироваться человеком. Для оценки случайности таких систем мы рассмотрим энтропию Колмогорова-Синая (взаимосвязанную с показателем Ляпунова и алгоритмической сложностью) и увидим, что детерминированный хаос может порождать алгоритмически случайные последовательности. Более того, в смешивающей системе, выборка $x_n, x_{n+k}, x_{n+2k}, x_{n+3k} \dots$ является асимптотически ($k \to \infty$) случайной, то есть с увеличением k члены выборки будут все менее зависимы.

2.2 Теоретико-сложностный подход

2.2.1 Вычислительная машина Тьюринга

Будем пользоваться известными понятиями из теории вычислительной сложности [5, 66, 16]. *Базовая машина Тьюринга*[1] состоит из (1) полубесконечной ленты, (2) считывающей головки и (3) схемы управления с конечном числом состояний. Обозначим машину как

$$T = \langle S, \mathcal{A}, \Gamma, F, q_0 \rangle ,$$

где S — конечное множество состояний; $\mathcal{A}$ — конечный алфавит ленты (в нашем случае, $\mathcal{A} = \{0, 1\}$); Γ — конечное множество правил вида $\gamma : S \times \mathcal{A} \to S \times \mathcal{A} \times \{L, N, R\}$; $F \subseteq S$ — множество заключительных состояний; и $q_0 \in S$ — начальное состояние. Множество $\{L, N, R\}$ содержит управляющие команды для перемещения головки: влево (L), на месте (N) и вправо (P). *Конфигурацией машины* T называется тройка $\langle s, \alpha, i \rangle$, где $s \in S$ — текущие состояние машины, $\alpha \in \mathcal{A}^*$ — строка на ленте, и $1 \leq i \leq |\alpha|$ — позиция головки не ленте, отсчитываемая с левого конца ленты.

Инициализация машины происходит следующим образом: (1) «загрузка» входной строки $\alpha \subset \mathcal{A}^*$ в качестве ленты; (2) перемещение головки в крайне левое положение; и (3) установка начального состояние $s = q$. За один такт работы,

[1] Тьюриг, Алан Матисон (1912–1954) — английский математик.

в соответствии с правилами Γ, машина Тьюринга: (1) Изменяет текущие состояние s; (2) печатает символ ленты, замещая, то что там было написано; (3) сдвигает головку налево или направо. Говорят, что машина *распознает* строку α, если существует такая последовательность правил $\gamma_1, \gamma_2, \ldots, \gamma_i, \ldots, \gamma_m$, $\gamma_i \in \Gamma$ (где m — время работы машины), которая переводит машину из начального состояния s_0 в допустимое заключительное состояние $s \in F$. Машина *не распознает* строку, если она никогда не переходит в допустимое заключительное состояние (либо останавливается в $s \notin F$, либо зацикливается).

Базовая модель машины Тьюринга может быть расширена различными способами, например: (1) Полубесконечная лента заменяется на ленту, бесконечную с двух сторон; (2) Одномерная лента заменяется на n-мерный куб. Можно показать, что что расширенная машина может быть заменена эквивалентной базовой.

Языком $\mathcal{L}$ называется множество строк (слов) конечной длины на алфавите $\mathcal{A}$. Говорят, что машина распознает язык $\mathcal{L}$, если она распознает все слова $\alpha \in \mathcal{L}$ и не распознает $\beta \notin \mathcal{L}$.

Детерминированная машина Тьюринга имеет однозначные правила перехода $\gamma : S \times \mathcal{A} \to S \times \mathcal{A} \times \{L, N, R\}$, то есть не существует двух правил с одинаковой левой частью $S \times \mathcal{A}$. Напротив, *недетерминированные машины* могут иметь неоднозначные правила (несколько вариантами возможного перехода).

Если существует полином $p(l)$, который ограничивает сверху время работы машины ($m < p(l)$) в зависимости от входной строки длины l (для всех достаточно больших l), то машину называют полиномиальной. Языки, распознаваемые полиномиальными *детерминированными* машинами Тьюринга, образуют класс **P**, а языки, распознаваемые полиномиальными *недетерминированными* машинами — класс **NP**.

Вероятностная машина — это детерминированная машина Тьюринга, снабженная дополнительной лентой, на которой записывается случайное слово. Неформально говоря, машина имеет возможность подбрасывать монету и совершать то или иное действие в зависимости от случайного результата. Вероятностная машина Тьюринга распознает язык $\mathcal{L}$, если она выдает правильный ответ (« 1 — распознано» или «0 — не распознано») с вероятностью более чем $2/3$, где вероятность берется в соответствии с равномерным законом распределения случайных слов одинаковой длины.

Класс **BPP** содержит языки, распознаваемые вероятностными машинами Тьюринга с полиномиальным ограничением на время работы.

Помимо распознавателя строк, машина Тьюринга, может выступать и в роли генератора (при этом выходная строка записывается на ленту). Таким образом, программа (совокупность правил) является алгоритмом генератора, а начальная

конфигурация машины задает все выходную последовательность.

2.2.2 Алгоритмическая сложность

В 1964 году Андрей Николаевич Колмогоров предложил определить алгоритмическую сложность двоичной последовательности как длину кратчайшей программы для универсальной машины Тьюринга, способной генерировать эту строку (к такому же представлению о сложности независимо и почти одновременно пришли Г.Чайтин и Р.Соломонов).

Определение 4. Алгоритмическая сложность $K_M(\alpha)$ конечной последовательности (строки) $\alpha \in \{0,1\}^n$ по отношению к машине Тьюринга M есть длина $l(\pi)$ самый маленькой программы π, которая вычисляет это последовательность, то есть

$$K_M(\alpha) = \min_{\pi: M(\pi)=\alpha} l(\pi).$$

Колмогоров показал, что существует универсальная машина Тьюринга U, которая осуществляет вычисления, эквивалентные π на произвольной машине M, причем изменения программы π, необходимые для ее работы на U, зависят от M и не зависят от α. Следовательно, алгоритмическая сложность K_M по отношению к любой машине M связана со сложностью программы $K_U(S)$ на универсальной машине U неравенством

$$K_U(S) \leq K_A(S) + C_A, \tag{2.1}$$

где C_M — некоторая константа, независимая от α [24]. Таким образом, в дальнейших рассуждениях можно опускать привязку к конкретной машине, полагая $K(\alpha) = K_U(\alpha)$.

К сожалению, задача нахождения программы π минимальной длины (или доказательство того, что программы, меньше известной, не существует) оказывается пока практически неразрешимой. Тем не менее теоретические приложение такого определения очень важно. Сложность по Колмогорову предлагает единый подход к проблемам сжатия данных.

2.2.3 c-сжимаемость и алгоритмическая случайность

Строка α_n длины n несжимаема для некоторый константы c (*c-несжимаема*), если $K(\alpha_n) \geq n - c$. Несжимаемые строки ($c = 0$ или мало) называются алгоритмически непредсказуемыми или алгоритмически случайными.

2.2.4 Посимвольная сложность

Для бесконечной строки α_∞ (или соответствующего генератора) интересно рассмотреть посимвольную сложность, то есть предел

$$c(\alpha_\infty) = \lim_{n\to\infty} \frac{K(\alpha_n)}{n}. \tag{2.2}$$

В силу (2.1) посимвольная сложность $c(\alpha_\infty)$ инвариантна относительно выбора вычислительной машины. Очевидно, что при конечной сложности $K(\alpha_\infty)$ (например, для псевдослучайного генератора), c устремляется к нулю. Для истинно случайной последовательности, длина программы равна длине строки, следовательно, $c = 1$. Вообще говоря, $c > 0$ только в том случае, если в сложность генератора бесконечна. В хаотической системе это происходит тогда, когда, либо сложность начальных условий бесконечно велика (при детерминированном поведении), либо некоторая случайность попадает из вне в процессе работы генератора.

2.3 Теоретико-информационный подход

Криптоаналитик рассматривает криптосистему как некоторый источник информации. В процессе криптоанализа, он обращается к статистическим свойства шифротекста и пытается восстановить криптографический алгоритм (точнее, его параметры). В идеальных криптосистемах, шифротекст не отличается от случайной последовательности и не содержит никакой полезной информации для криптоаналитика.

Существуют два различных подхода к статистическому анализу последовательностей. В теории информации [78] мы говором о статистическим свойствах *всех* последовательностей, создаваемых источником (генератором).

Напротив, второй подход, изучает статистические свойства одной конкретной последовательности (строки). Он привел к теории алгоритмической сложности, затронутую в предыдущем разделе.

В случае эргодической криптосистемы оба подхода оказываются эквивалентными.

2.3.1 Истинная случайность

Плотность распределения (ПР) случайной строки α на множестве возможных исходов $\mathcal{L} = \{\alpha_j\}$, $\alpha_j \in \{0,1\}^*$ есть функция $\Pr : \mathcal{L} \to [0,1]$ для которой $\sum_{\alpha\in\mathcal{L}} \Pr(\alpha) = 1$. Для конечного языка $\mathcal{L}$, плотность распределения задана ко-

нечным множеством значений $\Pr(\alpha_j)$, которые интерпретируются как вероятность появления строки α_j.

Определение 5. Последовательность α называется истинно случайной или непредсказуемой, если для любой длины $n > 0$ и для любых подстрок $\beta_n, \gamma_n \in \alpha$, вероятность $\Pr(\beta_n) = \Pr(\gamma_n)$.

Важным свойством истинно случайной последовательности является полное отсутствие корреляций, то есть для любого символа $s_i \in \alpha$, условная вероятность $\Pr(s_i|s_{i-1}, s_{i-2}, \ldots) = \Pr(s_i)$. Другими словами, сколь угодно большие знания о предыдущих символах, не могут повысить вероятность успешного предсказания следующего символа.

2.3.2 Энтропия Шеннона

Клод Шеннон обобщил понятие энтропии, первоначально известное в термодинамике, на абстрактные задачи в теории передачи и обработки информации [78]. Энтропия является мерой количества информации, необходимой для однозначного определения состояния системы среди всех возможных. Другими словами, энтропия является мерой нашего незнания о системе. В теоретической криптографии она определяет степень неопределенности криптосистемы для внешнего наблюдателя.

Энтропия n-битовой последовательности α_n определена как

$$H_n = -\sum\nolimits_{\alpha \in \{0,1\}^n} \Pr(\alpha_n) \log \Pr(\alpha_n), \tag{2.3}$$

где $\Pr : \{0,1\}^n \to [0,1]$ — функция плотности распределения α на множестве n-битовых строк. Максимум энтропии H_n достигается, когда $\Pr(\alpha_n)$ является равномерной плотностью распределения (то есть строка α_n истинно случайна).

Условная энтропия h_n определяет среднее количество информации, поступаемое с $(n+1)$-ым символом, при условии, что n предыдущих заданы:

$$h_n = h_{n+1|n} = \begin{cases} H_{n+1} - H_n, & n \geq 1 \\ H_1, & n = 1 \end{cases}$$

Другими словами, h_n характеризует среднюю неопределенность при предсказании $(n+1)$-ого символа. Так как знания о предыдущих символах (состояниях системы) не могут увеличить неопределенность будущего, функция H_n не убывает и $h_{n+1} \leq h_n$.

Для стационарного источника информации, существует предел

$$h_{Sh} = \lim_{n\to\infty} h_n = \lim_{n\to\infty} \frac{H_n}{n}. \tag{2.4}$$

В случае, если последовательность α задана Марковским процессом k-ого порядка, то $h_n = h_{Sh}$ для всех $n \geq k$. Марковская последовательность k-ого порядка характеризуется тем, что текущий (i-ый) символ зависит только от k предыдущих символов, то есть

$$\Pr(s_i|s_{i-1}, s_{i-2}, \ldots) = \Pr(s_i|s_{i-1}, s_{i-2}, \ldots, s_{i-k}), \quad s_i \in \alpha.$$

2.3.3 Взаимосвязь энтропии и сложности

Интуитивно понятно, что чем выше энтропия системы, тем сложнее должно быть ее внутреннее устройство. Сложность системы определяется размером «внутренней программы» системы. Результат работы этой программы есть последовательность символов. Очевидно, что, чем больше возможных состояний системы и управляющих правил, тем больше программа и сложнее последовательность. Измеряя статистические свойства этой последовательности, мы можем рассчитать энтропию системы и оценить ее непредсказуемость. Таким образом, сложность и энтропия взаимосвязаны, как *причина—следствие*.

Когда статистические свойства одной последовательности совпадает с обобщенными свойствами всех последовательностей, порождаемых генератором (то есть система эргодична), мы можем говорить о количественном равенстве энтропии и сложности.

Можно показать, что

$$\lim_{n\to\infty} \frac{\langle K_n \rangle}{H_n} = \frac{1}{\ln 2}, \tag{2.5}$$

где $\langle K_n \rangle = \sum_{\alpha_n \in \{0,1\}^n} \Pr(\alpha_n) K(\alpha_n)$. Таким образом средняя сложность $\langle K_n \rangle$ асимптотически пропорциональна энтропии Шеннона с коэффициентом $\ln 2$. Формально, это результат можно получить из теоремы Шеннона-Макмиллина [29].

2.4 Энтропия и сложность хаотических систем

2.4.1 Разбиение пространства состояний и символьная динамика

Рассмотрим хаотическую систему $\mathcal{S} = \langle X, f \rangle$ с f-инвариантной вероятностной мерой μ (см. раздел 1.2). Пусть задано разбиение β пространства состояний X на m неперекрывающихся подмножеств, то есть

$$\beta = \{X_1, X_2, ..., X_m\} : \qquad \bigcup_{i=1}^{i=m} X_i = X, \quad X_i \cap X_j = \emptyset, \ \forall i \neq j.$$

Каждому подмножеству X_i приписывается уникальный идентификатор, символ $s_i \in \mathcal{A}$. Функция σ задает разбиение β и символьные название регионов:

$$\sigma(x) = \{s_i \in \mathcal{A} | x \in X_i\}.$$

Траектория $\phi(x_0)$, проходя через различные подмножества X_i, формирует символьную траекторию $\alpha = \{s_k\}$. Различные начальные условия, принадлежащие одному и тому же подмножеству приведут к различным символьным траекториям (которые будут в начале совпадать). Наука, изучающая поведение символьных строк в динамических системах с заданным разбиением, называется символьной динамикой.

2.4.2 Энтропия Колмогорова-Синая

Показатель экспоненты Ляпунова (раздел 1.2.3) дает первую количественную информацию о том, как быстро мы теряем способность предсказывать поведение системы с течением времени. В этом смысле, более полезную оценку может дать энтропия Колмогорова-Синая h_{KS} [9]. Энтропия h_{KS} учитывает разрешение (точность измерения) с которым мы можем наблюдать за системой.

Пусть разбиение $\beta = \{X_1, X_2, ..., X_m\}$ задает разрешающую спорность наблюдателя. Измеряя текущие состояния x, наблюдатель может определить только тот факт что $x \in X_i.$, символ $\sigma(X_i) : x \in X_i$.

Энтропия n-символьной траектории $\alpha_n \in \mathcal{A}^n$ на разбиении β задана

$$H_n^\beta = -\sum_{\alpha_n} \Pr(\alpha_n) \log_{|\mathcal{A}|} \Pr(\alpha_n),$$

где $\Pr(\alpha_n)$ — вероятность появления символьной подстроки α_n. Условная энтропия $(n+1)$-ого символа (при известной предшествующей траектории α_n) равна

$$h_n^\beta = h_{n+1|n}^\beta = \begin{cases} H_{n+1}^\beta - H_n^\beta, & n \geq 1 \\ H_1^\beta, & n = 1 \end{cases}$$

Энтропия разбиения β задана

$$h^\beta = \lim_{n\to\infty} h_n^\beta = \lim_{n\to\infty} \frac{1}{n} H_n^\beta.$$

Энтропия Колмогорова-Синая есть точная верхняя граница h^β на множестве всех возможных разбиений

$$h_{KS} = \sup_\beta h^\beta. \tag{2.6}$$

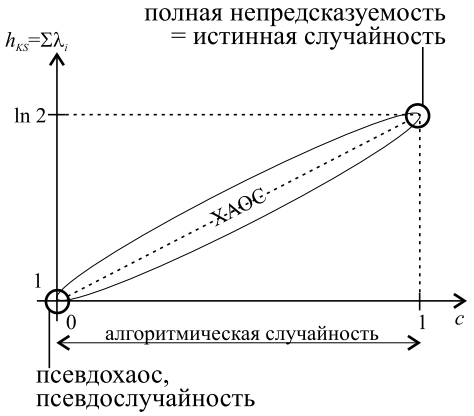

Рис. 2.3. Зависимость энтропии от сложности

Энтропия h_{KS} равна нулю для регулярных систем, положительна и конечна для для детерминированного хаоса и бесконечна для случайной величины. Энтропия h_{KS} связана с показателями Ляпунова ($h_{KS} = \sum_{1 \le d \le D} \lambda_d$) и пропорциональна интервалу времени T, на котором можно предсказать состояние хаотической системы [29].

2.4.3 Сложность траектории

Сложность траектории с началом в x_1 для открытого конечного разбиения β определена как

$$\mathcal{C}^{\beta}(x_1) = \limsup_{n \to \infty} \frac{1}{n} \min_{\alpha_n \in [\psi(x)]^n} K(\alpha_n),$$

где $[\psi(x)]^n = \left\{\alpha_n | f^{j-1}(x_1) \in X_j\right\}$ и $K(\alpha_n)$ — алгоритмическая сложность символьной строки.

Сложность траектории с началом в x_1 есть

$$\mathcal{C}(x_1) = \sup_{\beta} \mathcal{C}^{\beta}(x_1)$$

.

Определение 6. Траектория с началом в x_1 называется алгоритмически случайной, если она имеет положительную сложность

$$c(x_1) > 0$$

.

Теоремы А.Брудно и Г.Вайта определяют зависимость энтропии от сложности (рис. 2.3):

Теорема 1. алгоритмическая случайность траектории, [29, 58]) *Почти (в смысле меры μ) для всех $x \in X$, символьные траектории алгоритмически случайны и их сложность задана*

$$c(x) = \frac{h_{KS}}{\ln 2}, \tag{2.7}$$

Итак, несмотря на то, что определить алгоритмическую случайность заданной строки практически невозможно, почти все символьные строки конечного алфавита, порождаемые хаотическими системами, алгоритмически случайны.

Алгоритмическая случайность не является достаточным условиям для непредсказуемости. Даже если не существует детерминированного алгоритма, генерирующего последовательность, возможно существует вероятностная машина, которая предсказывает состояние с большей вероятностью успеха, чем случайных выбор. Формально, опишем вычислительную непредсказуемость в следующем разделе.

2.5 Псевдослучайность

2.5.1 Вероятностные ансамбли

Рассмотрим вероятностный ансамбль $\Pi = \{\Pr_i\}_{i\in\mathbb{N}}$, где $\Pr_i$ — плотность распределения строк на множестве $\{0,1\}^{l(i)}$, $l(i)$ — некоторый полином, задающий длину строки, и $i = \{1, 2, \ldots\}$. Ансамбль равномерных распределений $\Pi_0 = \{\Pr_{0,i}\}_{i\in\mathbb{N}}$ для любого $i \in \mathbb{N}$ и любых $\alpha, \beta \in \{0,1\}^i$ удовлетворяет равенству $\Pr_{0,i}(\alpha) = \Pr_{0,i}(\beta)$.

Для оценки «степени случайности» произвольной последовательности, необходимо сравнить ее вероятностный ансамбль с ансамблем равномерных распределений. Однако, на практике, обработать все подмножества длинной последовательности вычислительно сложно. Поэтому вводится понятие полиномиальной неотличимости. Вероятностные ансамбли полиномиально неразличимы, если они приписывают приблизительно одинаковые вероятности всем последовательностям, которые могут эффективно распознаны полиномиальной машиной Тьюринга:

Определение 7 (полиномиальная неотличимость, [89, 46, 58]). Пусть заданы вероятностные ансамбли $\Pi_1 = \{\Pr_{1,i}\}_{i\in\mathbb{N}}$ и $\Pi_2 = \{\Pr_{2,i}\}_{i\in\mathbb{N}}$, индексированные множеством $\mathbb{N}$. Рассмотрим вероятностную полиномиальную машину T, далее называемую *тестом*. На тест подается пара: индекс i и строка α. Обозначим $\Pr_1^T(i)$ вероятность того, что при поступлении на вход индекса i и строки α

в соответствии с распределением $\mathrm{Pr}_{1,i}$, тест T выдаст 1. Аналогично, $\mathrm{Pr}_2^T(i)$ есть вероятность того, что при поступлении на вход индекса i и строки α в соответствии с распределением $\mathrm{Pr}_{2,i}$, тест T выдаст 1. Тогда ансамбли Π_1 и Π_2 *не отличимы полиномом* $p(i)$, если для всех полиномиальных тестов T и досточно большого $i \in \mathbb{N}$ выполняется неравенство

$$\left|\mathrm{Pr}_1^T(i) - \mathrm{Pr}_2^T(i)\right| < \frac{1}{p(i)}.$$

Определение 8 (псевдослучайность, [89, 46, 58]). Вероятностный ансамбль $\Pi = \{\mathrm{Pr}_i\}_{i\in\mathbb{N}}$ называется *псевдослучайным*, если для любого положительного полинома $p(i)$, Π не отличим от ансамбля равномерных распределений $\Pi_0 = \{\mathrm{Pr}_{0,i}\}_{i\in\mathbb{N}}$.

Определение 9 (непредсказуемый вероятностный ансамбль, [89, 46, 58]). Пусть задан вероятностный ансамбль $\Pi = \{\mathrm{Pr}_{1,i}\}_{i\in\mathbb{N}}$. Пусть вероятностная полиномиальная машина T получает на входе индекс i и строку α, а выдает один бит, называемым *догадкой*. Пусть функция $bit(\alpha, r)$ возвращает r-й бит последовательности α, и $pref(\alpha, r)$ возвращает префикс из r бит, то есть $pref(\alpha, r) = bit(\alpha, 1)\, bit(\alpha, 2) \ldots bit(\alpha, r)$. Говорят, что машина T предсказывает следующий бит Π, если для некоторой полинома $p(i)$ и любого i,

$$\mathrm{Pr}\left(M\left(i, pref(\alpha, r)\right) = bit(\alpha, r+1)\right) \geq \frac{1}{2} + \frac{1}{p(i)}$$

где строка α выбирается случайно в соответствии с $\mathrm{Pr}_{1,i}$, а число r — в соответствии с равномерным распределением на множестве $\{0, 1, \ldots, l(\alpha) - 1\}$. Говорят, что ансамбль Π непредсказуем, если не существует вероятностной полиномиальной машины M, которая предсказывает следующий бит Π.

Теорема 2. [28, 46, 58] *Вероятностный ансамбль* Π *псевдослучаен тогда и только тогда, когда* Π *непредсказуем.*

2.5.2 Односторонняя функция

Односторонняя функция — основа многих криптографических систем. Неформально, односторонней функцией называется функция, которая быстро вычисляется в прямом направление ($\beta = f(\alpha)$), но не имеет эффективного алгоритма обратного преобразования т. е. инвертирования ($\alpha = f^{-1}(\beta)$). Разрыв между сложностью прямого и обратного преобразований определяет криптографическую эффективность односторонней функции.

Говорят, что односторонняя функция сохраняет длину (обозначается $1:1$), если битовая длина значения функции равна битовой длине аргумента. Такие

функции используются, например, в псевдослучайных генераторах. Если битовая длина значения односторонней функции постоянна при любой длине аргумента, то она называется хеш-функцией. Так, хеш-функция используется для хранения паролей или создания электронной подписи.

Определение 10 (односторонняя функция). Функция $f : \{0,1\}^* \to \{0,1\}^*$ называется односторонней, если

1) Существует детерминированная полиномиальная машина Тьюринга, которая выдает на выходе $f(\alpha)$ при поступлении на вход α;
2) Для любой вероятностной полиномиальной машины M, для любого полинома $p(n)$ и достаточно большого $n \in \mathbb{N}$

$$\Pr(M(f(\alpha), 1^n) \in f^{-}1(\alpha)) < \frac{1}{p(n)},$$

где строка α выбирается случайным образом на множестве $\{0,1\}^n$ в соответствии с равномерным законом распределения. Параметр 1^n ограничивает время работы машины M полиномом от длины искомого праобраза.

Мощным средством построения псевдослучайного генератора является крепкое ядро (hard-core) [28, 46, 58]. Предикат b функции $f(\alpha)$ называется крепким ядром, если он легко вычисляется, но сложно предсказуем из f(x).

Определение 11 (крепкое ядро). Пусть $f : \{0,1\}^* \to \{0,1\}^*$ и $f : \{0,1\}^* \to \{0,1\}$. Предикат b называется крепким ядром функции f, если

1) Существует детерминированная полиномиальная машина Тьюринга, которая выдает на выходе $b(\alpha)$ при поступлении на вход α;
2) Для любой вероятностной полиномиальной машины Тьюринга M, для любого положительного полинома $p(n)$ и достаточно большого $n \in \mathbb{N}$

$$\Pr(M(f(\alpha), 1^n) = b(\alpha))) < \frac{1}{2} + \frac{1}{p(n)},$$

где строка α выбирается случайным образом на множестве $\{0,1\}^n$ в соответствии с равномерным законом распределения.

Теорема 3. (существование крепкого ядра, [89, 65, 46]) *Если существует односторонняя функция, то существует одностороння функция с крепким ядром.*

2.5.3 Псевдослучайный генератор

Фундаментальная роль псевдослучайных генераторов в криптографии была затронута в разделе 1.1.4. Неформально, псевдослучайным генератором называется эффективный (детерминированный) алгоритм, который, получает на входе короткую последовательность (семя), и воспроизводит более длинную (обычно существенно более длинную) *псевдослучайную* последовательность.

Определение 12 (псевдослучайный генератор, [28, 46]). Пусть функция $l : \mathbb{N} \to \mathbb{N}$ удовлетворяет условию $l(n) > n$ для всех $n \in \mathbb{N}$. Псевдослучайный генератор с растягивающей функцией $l(n)$ есть детерминированный полиномиальный алгоритм G со следующими свойствами:

1) Для любой $\alpha \in \{0,1\}^*$ имеет место равенство $|G(\alpha)| = l(|\alpha|)$
2) Вероятностные ансамбли $\Pi = G(\mathrm{Pr}_0^n)$ и $\Pi_0^{p(n)}$ вычислительно неразличимы для достаточного большого n.

Теорема 4. (построение псевдослучайного генератора, [89, 46]) *Пусть f — односторонняя функция, сохраняющая длину, и предикат b — крепкое ядро f. Тогда $G(\alpha) = b(\alpha)b(f(\alpha))\ldots b(f^{l(|\alpha|)-1}(\alpha))$ есть псевдослучайный генератор с растягивающий функцией l.*

Теорема 5. (существование псевдослучайного генератора, [54, 50, 46]) *Псевдослучайные генераторы существуют тогда и только тогда, когда существуют односторонние функции.*

Следует отметить, что последовательность, порождаемая псевдослучайным генератором может иметь далеко не равномерную плотность распределения, однако, она будет вычислительно не отличима от равномерной плотности.

2.5.4 Проверка псевдослучайных генераторов

Классической задачей криптографии является оценка качества генераторов, считаемых псевдослучайными [57, 69, 74, 23]. Мы сформулировали определение псевдослучайного генератора, которое однако, невозможно полностью проверить на практике. Поэтому «измерение» псевдослучайности происходит экспериментально, при помощи набора тестов, выявляющих отклонения от равномерного распределения. Так, Национальный Институт Стандартов и Технологий (NIST) США рекомендует 16 тестов. Отметим наиболее важные из них:

Монобитовый тест. Количества единиц и нулей в последовательности должны быть приблизительно равны.

Частотный тест. Плотность распределения значений m-битовых блоков (часто, $m = 4$) должна казаться равномерной (проверяется с помощью распределения χ^2).

Тест пробегов. Пробегом будет называть подстроку, состоящую только из нулей или только из единиц $(0, 1, 00, 11, 000, 111, \ldots)$. Тогда длины пробегов должны соответствовать случайной последовательности, то есть среди всех пробегов половина должна иметь длину 1 бит, одна четверть должна иметь длину 2 бита, одна восьмая должна иметь длину 3 бита и так далее.

Далее будем использовать эти тесты для оценки псевдослучайности хаотических систем.

2.6 Псевдослучайный генератор на базе хаотической системы

Пусть динамическая система $\langle X, f\rangle$ обладает f-инвариантной вероятностной мерой μ и является хаотической и смешивающей (что подразумевает эргодичность). Рассмотрим разбиение пространства состояний X на два μ-равновероятных неперекрывающихся подмножества, то есть

$$\beta = \{X_0, X_1\} : \quad \mu(X_0) = \mu(X_1) = 1/2, \quad X_0 \cap X_1 = \emptyset$$

В каждый момент дискретного времени, состояние $x \in X_0$ соответствует символу «0», а состояние $x \in X_1$ — символу «1». Полученный генератор производит битовую последовательность α из начального состояния (семени) $x_0 \in X$. Обозначим генератор, как $G : X \to \{0,1\}^*$. Тогда

$$G(x) = \alpha = \{s_i\}_{i=1,2,\ldots}, \quad x \in X, \quad s_i \in \{0,1\}.$$

Щчепанский (Szczepankski) и Котульский (Kotulski) [82, 61] показали, что G может использоваться для генерации асимптотически случайных последовательностей.

Следующая теорема утверждает, что из различных начальных условий генератор производит различные последовательности с вероятностью 1:

Теорема 6. (уникальность траектории, [82]) *Для любого семени $x \in X$,*

$$\mu\Big(G^{-1}(\alpha)\Big) = 0$$

.

В силу эргодичности, число нулей в последовательности равно числу единиц (то есть будет выполнен монобитовый тест). Точнее, эргодическая теорема Биркгофа-Хинчина [9] может быть представлена в виде

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \chi_{X_0}\Big(f^i(x)\Big) = \int_{X_0} \chi_{X_0} d\mu = \mu(X_0),$$

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \chi_{X_1}\Big(f^i(x)\Big) = \int_{X_1} \chi_{X_1} d\mu = \mu(X_1),$$

где χ_{X_0}, χ_{X_1} — индикаторные функции X_0 и X_1. Так как $\mu(X_0) = \mu(X_1) = 1/2$, то среднее число нулей и единиц стремится к $n/2$.

Теорема 7. (монобитность, [82]) *Для любого семени $x \in X$, число единиц и нулей в последовательности α приблизительно равны.*

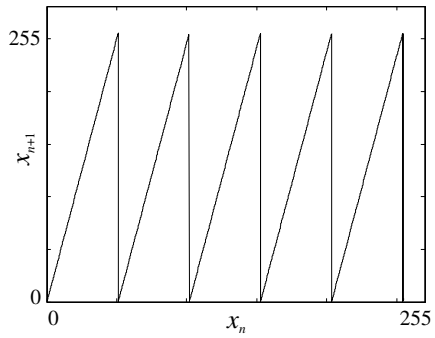

Рис. 2.4. Пилообразное преобразование для $p = 1279$ и $q = 255$.

Смешивающее свойство динамической системы обуславливает асимптотическую независимость битов последовательности:

Теорема 8. (асимптотическая независимость, [82]) *Для любого семени* $x \in X$ *и любого* $n = 1, 2, \ldots$ *выходные биты* $s_{(n-1)k}$ *и* s_{nk} *(рассматриваемые как случаемые величины) ассимтотически независимы с увеличением* k*, то есть*

$$\lim_{k\to\infty} \mu\Big(f^{-nk-k}(X_0) \cap f^{-nk}(X_1)\Big) = \mu\Big(f^{-nk}(X_0)\Big) \cdot \mu\Big(f^{-nk}(X_1)\Big)$$

и

$$\lim_{k\to\infty} \mu\Big(f^{-nk-k}(X_1) \cap f^{-nk}(X_0)\Big) = \mu\Big(f^{-nk}(X_1)\Big) \cdot \mu\Big(f^{-nk}(X_0)\Big).$$

Таким образом, с увеличением k, автокорреляционная функция последовательности $s_k, s_{2k}, \ldots, s_{nk}, \ldots$ будет стремится к дельта-функции. Композиция из k преобразований f выступает в роле однонаправленной функции: за счет сложности итерационного применения нелинейного преобразования (с потерей информации) восстановить x_{nk} из $x_{(n-1)k}$ становится все сложнее с увеличением k. Задача предсказания еще более усложняется, если управляющий параметр преобразования f неизвестен.

Разбиение пространство состояний $\beta = \{X_0, X_1\}$ выполняет функцию крепкого ядра и преобразует состояния системы (непрерывное) в биты. «Крепкое ядро» может быть задано отдельно (в явном виде), как будет показано ниже в примере.

Пример

Рассмотрим пилообразное преобразование (рис. 2.4)

$$x_{n+1} = rx_n \bmod q,$$

где $x_0 \in [0, q]$, $r = p/q > 1$, а p и q взаимопростые числа. Преобразование является хаотическим для всех r и имеет $\lambda = h_{KS} = \log r > 0$. Похожее дискретное преобразования широко используется в обычной криптографии. Классический линейный конгруэнтный генератор (Linear Congruental Generator, LCG) задан итерационной функцией

$$x_{n+1} = (Ax_n + C) \bmod M,$$

где $x_n \in \{0, 1, \ldots, M\}$ и A, C, M — фиксированные целые, выбираемые разработчиком [57]. Легко заметить известное свойство хаоса — растягивание и сжатие: в процессе преобразования, состояние системы сначала сильно растягивается (отображается на длинную прямую, например, перемножением или возведением в степень), а потом снова сжимается во множество допустимых состояний (операций mod).

Применим к каждому элементу хаотической последовательности $\{y_n\}_1^\infty$ некоторую периодическую функцию $H : \mathbb{R} \to [0, 1]$ с периодом 1. Пусть, например, $H(x) = \sin^2(\pi x)$. Тогда последовательность

$$\{y_n\}_1^\infty = \{y_1 = H(x_1), y_2 = H(x_2), y_3 = H(x_3) \ldots\}$$

тоже является хаотической [58]. Функция H играет роль крепкого ядра, хешируя x_i в y_i.

Для больших p и q, последовательность $\{y_n\}_1^\infty$ *непредсказуема в один шаг*: для любого $y_n \in \{y_i\}_1^\infty$ вероятность следующего элемента y_{n+1} равномерно распределена среди q вариантов, и вероятность предыдущего элемента y_{n-1} равномерно распределена среди p вариантов [58].

Получить битовую последовательность из $\{y_n\}_1^\infty$ можно путем разбиения X на два равновероятных подмножества. Определим функцию разбиения

$$\sigma(x) = \begin{cases} 0, & x \in X_0 = (0, 1/2] \\ 1, & x \in X_1 = (1/2, 1) \end{cases}$$

Имеем выходную последовательность генератора

$$s_1 = \sigma(y_1), \quad s_2 = \sigma(y_2), \quad s_3 = \sigma(y_3), \ldots$$

Таким образом, задавая хаотическое преобразование f с управляющими параметрами, хеширующее преобразование H и функцию разбиения σ, мы можем строить псевдослучайные генераторы.

Глава 3

Практическое приложение

В главе представлен обзор криптографических систем, построенных на базе компьютерной аппроксимации непрерывного и бинарного хаоса (так называемого псевдохаоса), а так же даны некоторые оценки их криптостойкости.

3.1 Хаос и псевдохаос

В предыдущей главе мы рассмотрели модель псевдослучайного генератора, построенного на базе хаотической системы, и увидели, что последняя может порождать бесконечные алгоритмически случайные последовательности. Более того, выборки из этих последовательностей могут быть асимптотически случайными. Однако, до сих пор речь шла о хаотических системах с бесконечным числом состояний. Практически, реализовать такую систему можно при помощи аналогового устройства. Так, теория непрерывного хаоса имеют большой потенциал для аппаратного кодирования и шифрования информации [7, 3, 32, 60, 34, 88, 80], например, в радио связи. В книге М. П. Кеннеди и соавторов *Хаотическая электроника в телекоммуникациях* [56] представлены теоретические основы и протоколы, готовых для промышленного использования. В настоящей работе главным объектом исследования являются приложения теории хаоса в компьютерной криптографии, поэтому аналоговые схемы не рассматриваются.

Легко понять, что хаотическая система не может быть реализована на машине с конечном числом состояний. Каждое последующие состояние системы не должно совпадать с каким-либо предыдущим состоянием траектории. В противном случае (например, в результате округления) траектория превращается в циклическую орбиту. Все компьютерные модели хаоса являются аппроксимацией (приближением) математического хаоса. Аппроксимация в той или иной степени передает свойства исходной системы на начальных итерациях, но в пре-

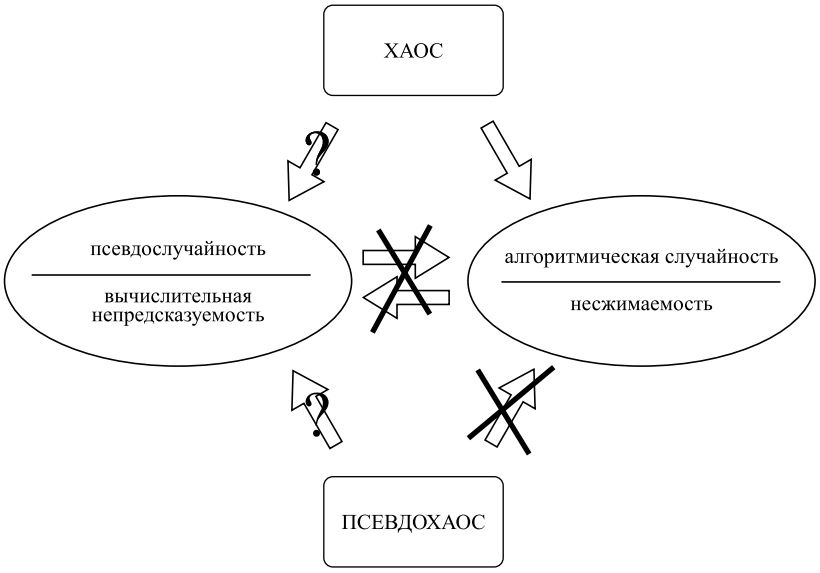

Рис. 3.1. Свойства хаотической и псевдохаотической систем.

деле $n \to \infty$ дает ассимтотически неправильное приближение (то есть траектория модели расходится с траекторией исходной системы). Будем называть компьютерные аппроксимации **псевдохаосом**.

Поведение псевдохаотической системы может качественно отличаться от исходной хаотической системы. Следовательно, сразу переносить свойства о непредсказуемости хаотической системы на ее аппроксимацию будет неверно.

Когда псевдохаотическая система порождают вычислительно непредсказуемые (псевдослучайные) последовательности (рис. 3.1)? Универсального ответа на этот вопрос пока нету, как, впрочем, и для хаотических систем. На сегодняшний день, псевдослучайные генераторы обычно испытываются при помощи тестов (раздел 2.5.4).

3.1.1 Длина периода

При построении псевдохаотических криптосистем важно знать минимальную и среднюю длину циклической орбиты. Разные классы криптосистем предъявляют разные требования к длине орбиты:

- Орбиты со короткой длиной крайне опасны так как ведут к образованию паттернов (рис. 3.2-а).
- Для потоковых шифров и псевдослучайных генераторов длинных после-

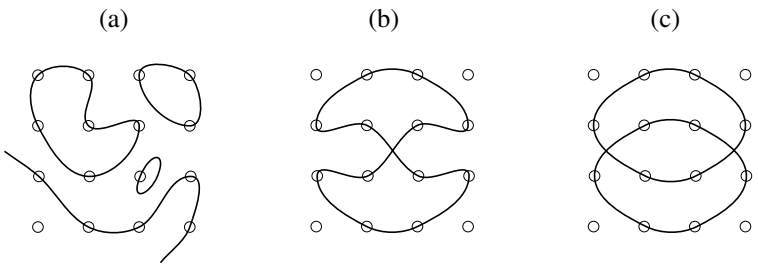

Рис. 3.2. Орбиты псевдохаотических систем. (a) Короткие и непредсказуемые орбиты (не подходит для криптографии); (b) Одна длинная орбита (подходит для потокового шифра); (c) Несколько однотипных орбит (подходит для блочного шифра).

довательностей идеальна одна длинная орбита проходящая через все пространство состояний (рис. 3.2-b).

- Для блочных шифров могут подойти ансамбль однотипных орбит (рис. 3.2-c). Расположение таких орбит в пространстве должно зависеть от ключа сложным образом; в противном случае, сужается пространство поиска открытого текста в процессе криптоанализа.

3.1.2 Экспонента Ляпунова

В псевдохаотических системах состояние системы кодируется конечным числом битов и, следовательно, имеет конечную точность σ. Поэтому экспоненциальное расхождение траекторий, определяемое соотношением

$$e^{n\lambda} = \frac{|f^n(x_0 + \varepsilon) - f^n(x_0)|}{\varepsilon}, \tag{3.1}$$

при $n \to \infty$, $\varepsilon \to 0$, будет на самом деле ограничено $\varepsilon \geq \sigma$. Таким образом, рост среднего значения дроби (3.1) остановится на некотором значении $\langle d \rangle / \varepsilon$, где $\langle d \rangle$ — среднее расстояние между текущем и начальным состоянием, то есть $d = |x - x_0|$. На рис. 3.3 показана типовая «экспонента» псевдохаотической системы: на первых итерациях наблюдается экспоненциальный рост, затем — линейное увеличение и выравнивание.

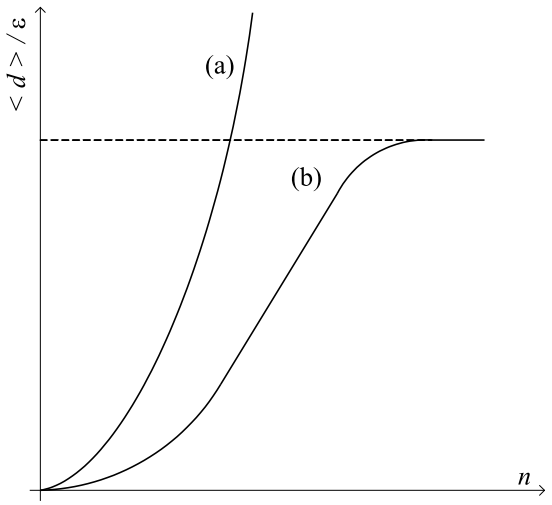

Рис. 3.3. Экспоненты Ляпунова в хаотической (a) и псевдохаотической (b) системах

3.2 Псевдохаос на базе математики с плавающей запятой

3.2.1 Общие свойства

Математика с плавающей запятой (МПЗ) [52] является наиболее простом подходом к моделированию непрерывных систем на современных ЭВМ. Представление числа с плавающей запятой позволяют хранить действительные числа в битовой строке с некоторой конечной точностью.

Действительное число x может быть записана как бесконечная десятичная дробь в двоичном представлении $b_m b_{m-1} \dots b_1 \, . \, a_1 a_2 \dots a_s$, где a_i, b_j — биты, $b_m b_{m-1} \dots b_1$ соответствует целой части, а $a_1 a_2 \dots a_s$ — дробной части числа.

При вычислениях с конечной точностью, вместо итерационной функции $x_{n+1} = f(x)$, мы можем записать

$$x_{n+1} = round_k \left(f \left(x_n \right) \right),$$

где $k \leq s$ и $round_k(x)$ есть функция округления, заданная как

$$round_k(x) = b_m b_{m-1} \dots b_1 \, . \, a_1 a_2 \dots a_{k-1} \left(a_k + a_{k+1} \right).$$

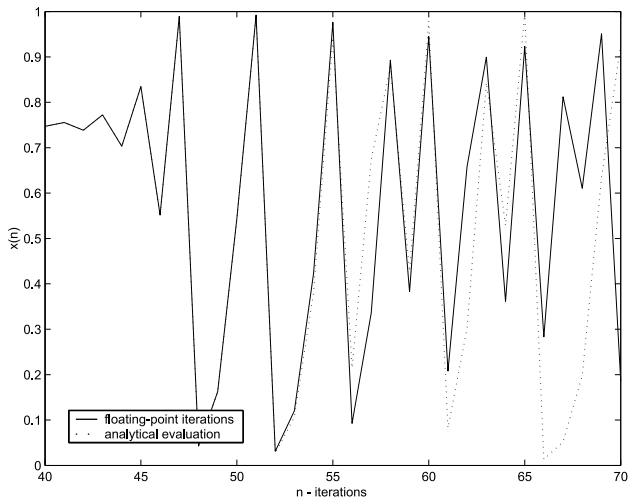

Рис. 3.4. Траектория непрерывной системы (логистическая парабола) и траектория аппроксимированной системы с точностью 64 бита. Ошибка округления усиливается на каждой итерации. Траектория непрерывной системы получена при помощи точного аналитического решения.

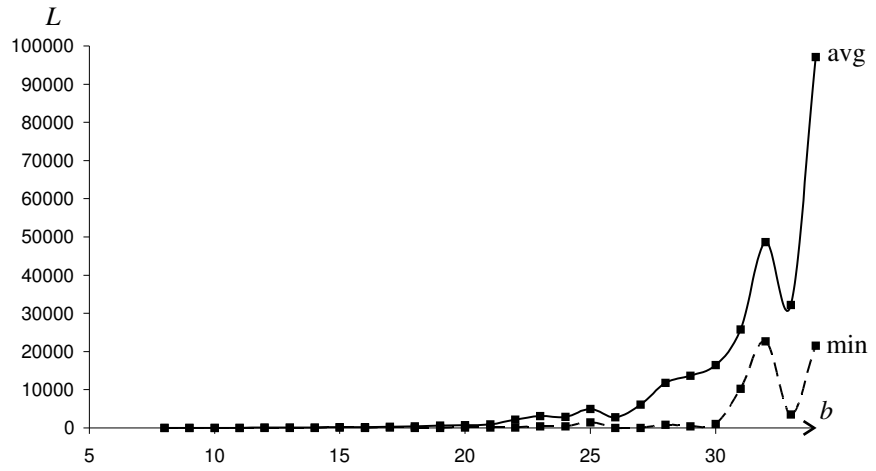

Рис. 3.5. Минимальная и средняя длины орбит (L_{min} и L_{avg} соответсвенно) в логистической системе в зависимости от точности вычисления b (в битах).

Одной из проблем моделирования непрерывных динамических систем является накопление ошибки округления. Функция $round_k(x)$ применяется на каждой итерации, и ошибка накапливается и усиливается из-за чувствительности системы к начальным условиям. Траектория исходной и аппроксимированной систем расходятся очень быстро. Например, на рис. 3.4 представлены временные ряды исходной и аппроксимированной системы. Как было обнаружено Лоренцем[1], «*. . . малая ошибка в прошлом приводит к огромной ошибке в будущем. Предсказание становится невозможным . . .*» Таким образом, МПЗ система не является корректным приближением непрерывной хаотической системы.

Помимо неправильного асимптотического поведения, аппроксимированные системы могут проявлять другие «опасные» свойства уже в начале траектории. Длина орбиты может быть непредсказуемо короткой и образовывать легко распознаваемый паттерн, что в криптографии неприемлемо. На рис. 3.2 (а) условно показаны типовые орбиты МПЗ-аппроксимации. Возможно, что в результате округления состояния, траектория безвозвратно покинет хаотический аттрактор и войдет в некоторые стационарного состояние. Так, во многих классических системах переменные, определяющие состояния системы, могут иметь бесконечно близкие, но не равные нулю значения; а в случае округления до нуля, хаотическое поведения прекращается. На рис. 3.5 представлены результаты численного моделирования при различных точностях вычислений. Как видно, средняя длина орбита непредсказуема, а минимальная остается малой даже при высокой точности.

Другая проблема с МПЗ-аппроксимациями связана с тем, что различные платформы (аппаратные и программные) используют различные алгоритмы вычисления математических функций и сохраняют промежуточные результаты с разной точностью. Так как хаотические генераторы крайне чувствительны к точности, очень вероятно, что хаотические алгоритмы шифрования, реализованные на различных платформах, окажутся несовместимыми.

Не смотря на перечисленные (очень существенные недостатки), немало исследователей пробовали и продолжают строить криптографические системы на базе МПЗ-аппроксимации непрерывной системы. Ниже представлен обзор хаотических систем и схем шифрования.

3.2.2 Разбиение пространства состояний

Числа с плавающей запятой имеют известный двоичный формат, обладающий некоторый избыточностью. Следовательно, направлять эту последовательность

[1]Лоренц, Эдвард, американский метеоролог, исследовал модель атмосферных явлений и открыл стабильный хаотический аттрактор в 1960 г.г.

сразу на выход генератора небезопасно. Часто используют хеширующее преобразование или и просто функцию разбиения (раздел 2.4.1), $\sigma : X \to \{0,1\}^m$, где X — множество чисел в формате с плавающей запятой. Простейший пример — взятие последнего значащего бита.

Таким образом, пространство X разбивается на два равновероятных подмножества X_0 и X_1, которые преобразуются функцией σ в выходной бит 0 или 1. Условие полного покрытия $X = X_0 \cup X_1$ здесь не обязательно (то есть некоторым состоянием $x \notin X_0 \cup X_1$ может соответствовать пустой символ). Более того, задавая особые разбиения мы можем получать различные статистические свойства выходной последовательности или не использовать ту область пространства состояний, которая не удовлетворяет криптографическим требованиям (рис. 3.12).

Число подмножеств X_i может быть больше 2 (например, $4, 8, 16 \ldots$). Тогда за одну итерацию генератор будет производить не один, а несколько псевдослучайных битов ($m = 2, 3, 4 \ldots$). Очевидно, что увеличение m понижает криптостойкость генератора, так как предсказать аргумент «крепкого ядра» σ становится легче.

3.2.3 Преобразование Чебышева

В 1983 Эрбер *et. al.* [38] предложил использовать смешивающий полином Чебышева для моделирования случайных процессов на цифровых (дискретных) ЭВМ. Итерационная функция, построенная на базе полинома Чебышева, имеет вид

$$x_{n+1} = x_n^2 - 2, \qquad x_n \in (-2, 2)\,. \tag{3.2}$$

Временной ряд $x_0, x_1, x_2, \ldots$ является хаотическим, однако, некоторые значения x_n должны избегаться ($x_n \neq -1, 0, 1$). Теоретически, для любого нецелого значения $x_0 \in (2, 2)$, траектория не войдет в запрещенное состояние, что нельзя сказать о МРЗ-аппроксимации. Так, при точности в 40 бит средняя длина циклической орбиты составляет 105 итераций.

Полученный временной ряд $\{x_n\}$ обладает не лучшими криптографическими свойствами. В частности, плотность распределения $\Pr(x_n)$ не является равномерной, как показано на рис. 3.12. Мы можем попробовать изменить статистические свойства последовательности, применив дополнительное преобразование

$$y_n = \frac{4}{\pi} \arccos\left(\frac{x_n}{2}\right) - 2. \tag{3.3}$$

Функциональное преобразование (3.3) случайной величины x_n по форме совпадает с функцией распределения $\Pr(x_n < x)$. Поэтому последовательность функция распределения последовательности y_n быть близка к прямой. На рис. 3.6

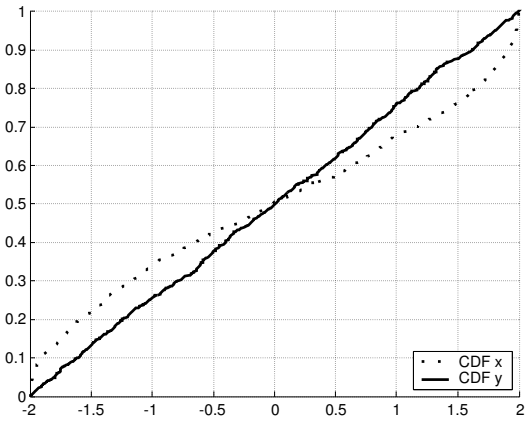

Рис. 3.6. Эмпирическая функция распределения временного ряда, полученного преобразованием Чебышева, до (точечная линия) и после (сплошная линия) выравнивания. Сплошная линия похожа на равномерный закон распределения на отрезке $(-2, 2)$.

точечный линей показана функция $\Pr(x_n < x)$, а сплошной линей — функция $\Pr(y_n < y)$.

Однако, последовательность y_n не является псевдослучайной, не смотря на то, что график функции распределения соответствует равномерному закону. Функциональное преобразование (3.3) не позволяет изменить *порядок* элементов в последовательности x_n, которые взаимосвязаны детерминированным преобразованием (3.3). То есть условная вероятность $\Pr(x_n|x_{n-1}, x_{n-2}, \ldots)$ остается инвариантным и позволяет эффективно предсказывать последовательность y_n. Впрочем, зависимость между элементами последовательность может быть уменьшена путем удаления промежуточных членов (раздел 2.6).

Такие недостатки преобразования Чебышева для криптографических приложений были отмечены некоторыми исследователями (например, [53]).

3.2.4 Логистическая парабола

Известная логистическая парабола эквивалентна преобразованию Чебышева. Ранее, в 1976, Митчелл Фейгенбаум (Mitchell Feigenbaum) исследовал сложное поведение так называемой логистической системы. Итерационная функция задана отношением (рис. 3.7)

$$x_{n+1} = 4rx_n\left(1 - x_n\right), \tag{3.4}$$

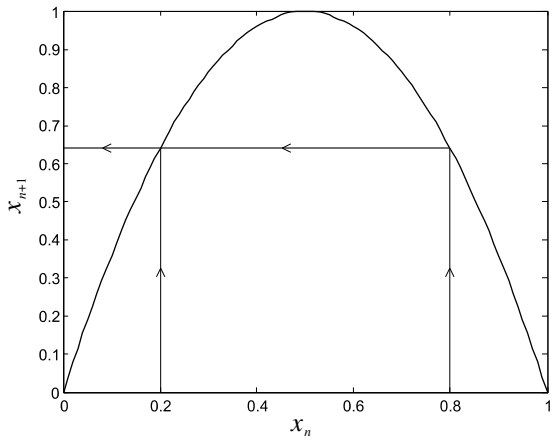

Рис. 3.7. Логистическая парабола при $r = 0.99$.

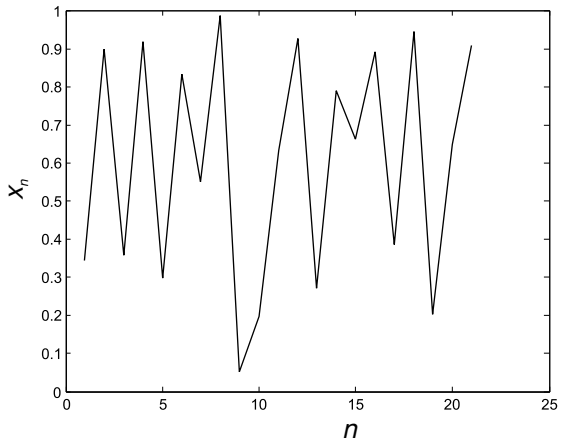

Рис. 3.8. Временной ряд полученной в логистической системе при $x_0 = 0.34$ и $r = 0.99$.

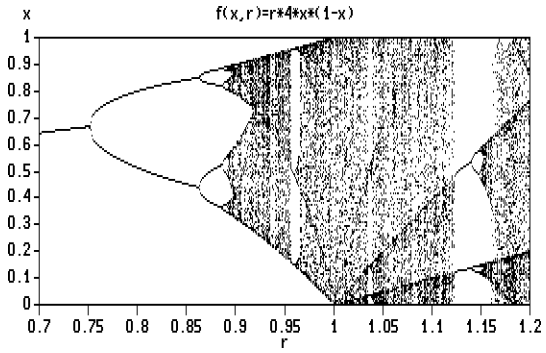

Рис. 3.9. Бифуркация логистической системы. Наиболее непредсказуемое поведение наблюдается при $r = 1$.

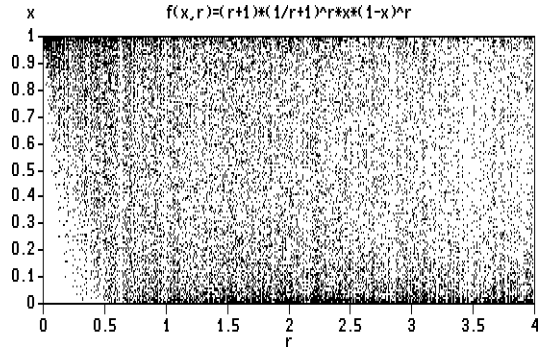

Рис. 3.10. Бифуркация хаотической системы Мэтью.

где $x \in (0, 1)$ и $r \in (0, 1)$.

Для некоторых значений управляющего параметра r, система является хаотической. Показатель экспоненты Ляпунова для траектории с началом в x_0 определяется выражением

$$\lambda(x_0) = \frac{1}{N} \sum_{n=1}^{N} \log |r(1 - 2x_n)|.$$

Среднее численное значение, например, для $r = 0.9$ есть $\lambda(0.5) \approx 0.7095$.

Было замечено, что система порождает временной ряд, который *кажется* псевдослучайной (рис. 3.8). Бифуркационная диаграмма Фейгенбаума (рис. 3.9) показывает предельные значения x_n при $x \to \infty$ для различных значений па-

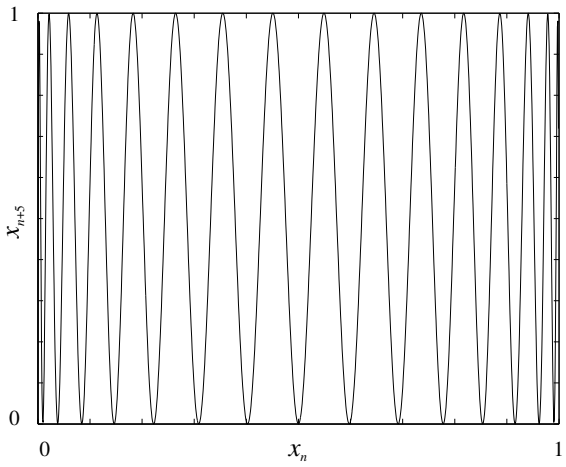

Рис. 3.11. Решение логистической системы для $n = 5$.

раметра r. С увеличением r число точек аттрактора увеличивается с 1 до 2, 4, 8 и так далее до бесконечности. Предполагалось, что в области $r \to 1$ система непредсказуема, то есть при неизвестном точном значении r и достаточно большом n невозможно предсказать x_n из x_0, и, наоборот, из x_n — восстановить x_0. Сегодня известно, что решение для любой траектории логистической системы может быть записано в аналитическом виде [61]. Так, для $r = 1$

$$x_n = \sin^2\left(2^n \arcsin\sqrt{x_0}\right). \tag{3.5}$$

При $n = 1$ имеем исходное преобразование (3.4).

Таким образом, состояние x_n может быть рассчитано из x_0 без проведения n итераций. На рис. 3.11 представлено значения $x_5 = f^5(x_0)$. С увеличением n будет возрастать число пиков, наглядно показывая суть чувствительности к начальным условиям. Важное приложение систем с точным аналитическим решением будет рассмотрено в разделе 3.2.8.

Другой особенностью МПЗ-аппроксимации нелинейной системы с действительным состоянием является существование совпадающих траекторий с разными начальных состояниями. На рис. 3.7 видно, что, например, $x_n = 0.2$ и $x_n = 0.8$ отображаются в одно и тоже значение x_{n+1}. В системе с бесконечном числом состояний, вероятность такого совпадения бесконечно велика. Напротив, в системе с округлением, вероятность совпадения части (конца) траекторий становится существенной, что уменьшает эффективное пространство ключей.

Бианко *et al.* [27] предложили использовать логистическое преобразование (3.4) для генерации последовательности чисел в формате с плавающей запятой.

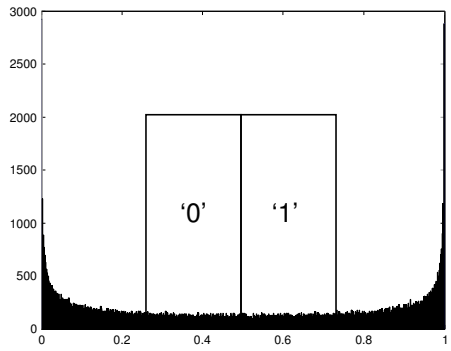

Рис. 3.12. Плотность распределения состояний в логистической системе. Неполное разбиение X на подмножества $X_0 \cup X_1 \subset X$ позволяет использовать только ту область пространства состояний, которая удовлетворяет требования криптографического приложения.

Эта последовательность преобразуется в поток битов при помощи разбиения пространства на два равновероятных подмножества, как показано на рис. 3.12. Полученная битовая последовательность складывается по модулю 2 (XOR) с исходным текстам (см. шифр Вернама, раздел 1.1.3). Начальные условия (состояние x_0 и параметра r) представляют собой секретный ключ.

Как уже было отмечено, число единиц и нулей в такой последовательности теоретически совпадает, но *порядок* элементов совсем не псевдослучайный и предсказуемый. Более того, длины орбит для некоторых ключей могут быть неприемлемо короткими [84, 55].

Мэтью [68] расширил логистическое преобразование для криптографических задач:

$$x_{n+1} = (r+1)\left(\frac{1}{r}+1\right)^r \cdot x_n \left(1-x_n\right)^r, r \in (1,4)\,.$$

Аттрактор системы Мэтью имеет бесчисленное множество состояний для широкого диапазона управляющего параметра r (рис. 3.10). Таким образом, существенно увеличивается пространство ключей $\langle x_0, k\rangle$. К сожалению, остаются другие немаловажные проблемы, свойственные МПЗ-генераторам (короткие орбиты, сильная корреляция).

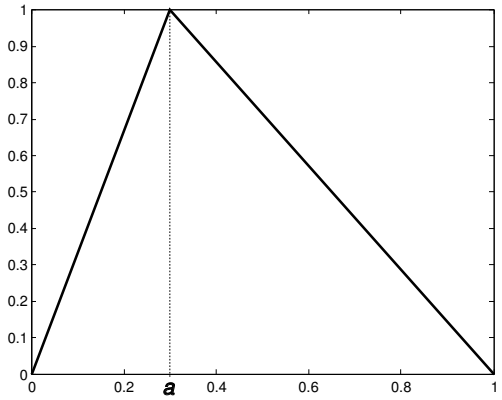

Рис. 3.13. Палаточное преобразование.

3.2.5 Палаточное преобразование

Одномерное палаточное преобразование задано функцией

$$x_{n+1} = \begin{cases} a \cdot x_n, & 0 \le x_n \le a \\ \frac{1-x_n}{1-a}, & a < x_n \le 1 \end{cases}, \tag{3.6}$$

где параметр a задает абсциссу верхней точки палатки (рис. 3.13).

Показатель экспоненты Ляпунова зависит от параметра a и определяется соотношением $\lambda(a) = -a\ln(a) - (1-a)\ln(1-a)$ почти для всех $x_0 \in (0,1)$ [64]. Численно, $\max\limits_{a\in(0,1)} \lambda(a) \approx 0.693$ при $a = 0.5$.

Палаточная система имеет аналитическое решение [61]

$$x_n = \frac{1}{\pi}\arccos\left(\cos 2^n \pi x_0\right). \tag{3.7}$$

Хабутсу *et al.* [49] предложили блочную схему шифрования (рис. 1.3) на базе перевернутого палаточного преобразования

$$x_{n+1} = \begin{cases} a \cdot x_n, & r_n = 0 \\ (a-1)\,x_n + 1, & r_n = 1 \end{cases}, \qquad n \in [1, N]. \tag{3.8}$$

где $r_i \in \{r_i\}_1^N$ и $a \in [0.4, 0.6]$. Секретный ключ состоит из параметра a (в формате с плавающей запятой) и N-битовой строки $\{r_i\}_1^N$. Шифрование происходит путем N-кратного ($N = 75$) преобразование блока открытого текста в шифротекст. Размер блока открытого текста — 64 бита, шифротекста — 147 бит. Таким

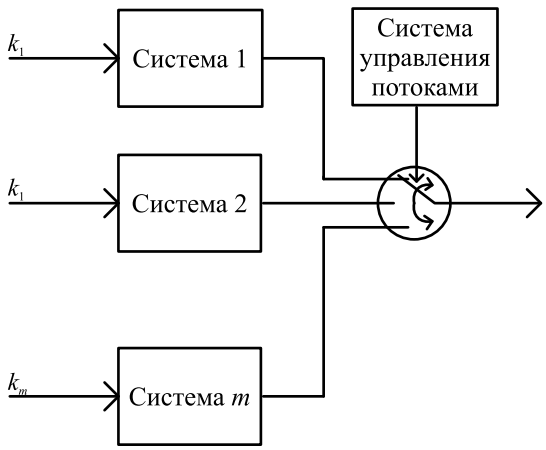

Рис. 3.14. Многопоточная схема шифрования

образом, N криптографических преобразований обеспечивает 2^N возможных вариантов шифрования одного блока текста.

Эли Байем [26] показал, этот шифр может быть легко расколот при атаке с выбранным открытым текстом, а сложность атаки при известном открытом тексте составляет $K = 2^{38}$.

3.2.6 Многопотоковый шифр

Протопопеску [72] предложил использовать m разных хаотических систем для поточного шифрования. Множество начальных условий для m систем являются ключом. Из каждой системы поступает число x_n в формате с плавающей запятой, из которого извлекается один бит b_n, после чего все m битов суммируются в один бит b операцией XOR. Полученный бит b подается на выход генератора. Процесс повторяется для генерации «одноразового блокнота» достаточной длины.

Метод Протопопеску может быть обобщен следующим образом:

1) Генерация битов каждой системы может происходить с пропусками итераций. Другими словами, хаотическая система может подавать выходной бит только каждую q-ую итерацию. Это уменьшает (существенно) видимые корреляции между элементами последовательности (раздел 2.6).
2) Набор хаотических систем может быть нефиксированным и определяться сеансом связи (или ключем).
3) Системы могут чередоваться в соответствии с некоторым секретным пра-

вилом, зависимым от ключа (рис. 3.14).

С учетом обобщения 1, криптографическую систему Протопопеску можно записать в виде системы

$$\begin{cases} x^1_{n+1} = f_1(x^2_n, k^1), & b^1_j = \sigma_1(x^1_{qj}) \\ x^2_{n+1} = f_2(x^2_n, k^2), & b^2_j = \sigma_2(x^2_{qj}) \\ \cdots & \cdots \\ x^m_{n+1} = f_m(x^m_n, k^m), & b^m_j = \sigma_m(x^m_{qj}) \end{cases}$$

$$b_j = b^1_j \oplus b^2_j \oplus \ldots \oplus b^m_j,$$

где
$f_1, f_2, \ldots, f_m$ — итерационные функции, используемые в текущем сеансе связи; $\langle x^1_0, k^1, x^2_0, k^2, \ldots, x^m_0, k^m \rangle$ — начальные условия; $b^1_j, b^2_j, \ldots b^m_j$ — выходные биты систем в $(n = qj)$-ой момент времени; b_j — выходной бит генератора.

Псевдослучайный генератор со свойствами (1)-(3) был реализован и исследован в рамках этой работы. Для $m = 5$ и $q > 20$ выходная последовательность становится неотличимой от случайной. Тем не менее решение громоздко и не дает качественного решения перечисленных выше проблем МПЗ. В частности, наблюдаются «отключения» систем в результате округления в ноль. Это не приводит к тривиальному поведения всей системы из-за «m-кратного резервирования», но является явным признаком неэффективности генератора.

3.2.7 Другие хаотические шифры

Гелафер *et. el.* [45] разработал хаотический потоковый шифр на основе преобразования

$$f(x) = \left(a + \frac{1}{x}\right)^{\frac{x}{a}} \qquad x \in (0, 10), \quad a \in [0.29, 0.40].$$

Ключом является комбинация начального состояния x_0 и параметра a. После $n_0 = 200$ начальных итераций, система шифрует байт открытого текста p_1 в МПЗ-шифротекст $c_1 = f^{n_0+p_1}(x_0)$ (32,64,128 бит), то есть преобразование f применяется еще $(p_1 \in [0, 255])$ раз. Последующие байты открытого текста шифруются с помощью той же траектории (рис. 1.4-c):

$$c_i = f^{n_0+\sum_{k=1}^{i} p_i}(x_0).$$

Недостатками такой схемы шифрования является: (1) существенное (8 − 10-ми кратное) увеличение размера шифротекста по сравнению с открытом текстом; (2) «ненадежные» траектории, аналогично другим криптосистемам на базе МПЗ.

Баптиста [25] и Вонг [87] предложили несколько методов шифрования, в которых открытый текст кодируется числом итераций. Пространство состояний X разбивается на m неперекрывающихся подмножеств $\{X_1, X_2, ..., X_m\}$ (покрывающих X полностью или частично). Каждому подмножеству приписывается уникальный символ открытого текста. После инициализации некоторым начальным условием, которое является ключом, система производит n_0 итераций. Далее все символы открытого текста преобразуются в последовательность целых чисел, равных числу итераций для достижения соответствующего подмножества:

$$c_i = n_i - n_{i-1}, \quad i = 1, 2, \ldots,$$

где $p_i = \sigma(X_i)$, $f^{n_i} \in X_i$. На рис. 1.4-b показана траектория криптосистемы со счетчиком итераций. Эта базовая схема шифрования может быть расширена путем введения дополнительных параметров (например, минимальное число итераций).

Очевидно, что рассмотренная криптосистема обладает дополнительно к недостатком МЦЗ-шифров еще следующими нежелательными качествами: (1) имеется ненулевая вероятность переполнения счетчика $c_i = n_i - n_{i-1}$, так как он имеет конечную длину; (ii) шифротекст в 2-3 раза больше, чем исходный текст.

Хо продолжил исследование метода Баптисты и Вонга и оценил его быстродействие и криптостойкость. Оптимальный размер для блока открытого текста — 4 бита, а блока шифротекста — 10 бит. В [51] Хо обсуждает практические аспекты хаотической криптографии: способы разбиения, распределение траекторий и алгоритмическая сложность.

Котульский *et el.* [62, 63] исследовали общие приложение теории в хаоса в криптографии и подчеркнул важность смешивающего свойства для криптографических систем. Котульский рассмотрел криптографическую схему, в которой шифрование осуществляется m-кратным применение обратного преобразование f^{-1} (хаотического и смешивающего), а дешифрование — m-кратным прямым преобразованием f. Ключ вводится в начальные условиях x_0 и k. Вычисление f и f^{-1} осуществляется при помощи МПЗ. В качестве примера хаотической системы Котульский рассматривает процесс отражения геометрического луча в квадрате. Система обладает тремя размерностями — две координаты и угол, определяющий направление луча.

Для увеличения непредсказуемости (числа состояний, нелинейности, сложности) можно использовать системы дифференциальных уравнений более высокого порядка и размерности. Например, Паар [71] предложил использовать

модель робота для шифрования данных

$$m\frac{d^2x}{dt^2} - \beta_2\left(\frac{dx}{dt}\right)^2 sign\left(\frac{dx}{dt}\right) - \gamma_2 sign\left(\frac{dx}{dt}\right) - \delta_{21}x - \delta_{23}x^2 = \\ = L\frac{\omega_0^2}{2\pi}\cos(\omega_0 t) - \zeta_{21}e^{\lambda_{21}t} - \zeta_{22}e^{\lambda_{22}t},$$

которая соответствует жесткой пружине с коэффициентами трения δ_{21} и δ_{23}. Правая часть представляет собой периодическую силу, зависимую от времени с амплитудой $L\left(\omega_0^2/2\pi\right)$ и обратной силой, заданной соответствующими параметрами.

На текущей момент, ни одной такой системы не реализованы в виде рабочего алгоритма шифрования (по крайне мере не известно автору). Это объясняется, главным образом, сложным численным интегрированием и неравномерным распределением системных переменных.

3.2.8 Системы с неоднозначным преобразованием и точным решением

В предыдущих разделах мы уже рассмотрели точные решения для логистической и палаточной систем. Точное (аналитическое) решение позволяет вычислить произвольный член x_n последовательности сразу из начальных условий x_0, минуя n-кратное вычисление итерационной функции. Точное решение позволяет избежать накопление ошибки и существенно повышает скорость вычислений. На рис. 3.4 видно, как различаются временные последовательности логистической системы, полученные при помощи итерационных и точных вычислений.

Большинство хаотических систем, рассмотренных применительно к криптографии, имеют точное решение. Однако этот факт не использовался в схемах шифрования и, возможно, даже не был известен разработчикам. Ясно, что система становится небезопасной, если цепочку криптографических итераций можно заменить одним простым выражением.

Точное решение для динамической системы может быть записано в форме [47, 61]

$$x_n = \Psi\left(\theta T \kappa^n\right), \tag{3.9}$$

где $\Psi(t)$ — переодическая функция с периодом T; κ — целое число; и θ — некоторый действительный параметр, определяющий начальные условие из отношения

$$x_0 = \Psi\left(\theta T\right).$$

Система является хаотической, если $\lambda = \ln\kappa > 0$.

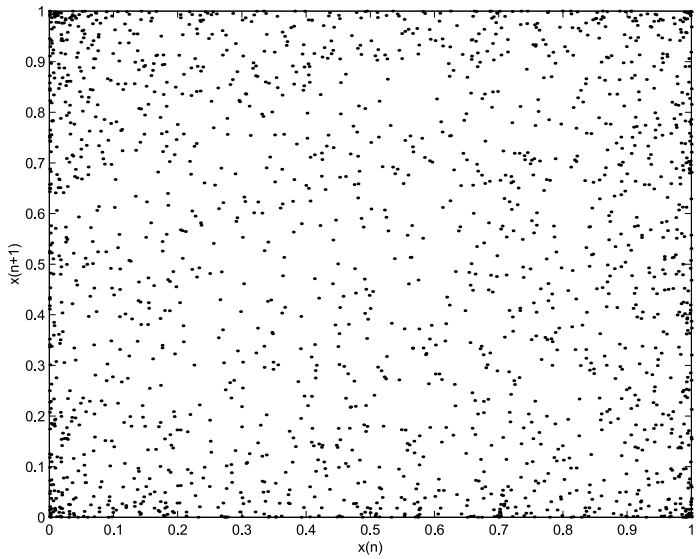

Рис. 3.15. Неоднозначное преобразование $x_{n+1} = f(x_n)$

Рассмотрим систему с точным решением, заданную выражением

$$x_n = \sin^2(\pi\theta\kappa^n). \tag{3.10}$$

Преобразование $f : x_n \to x_{n+1}$ для такой системы определяется отношением

$$x_{n+1} = \sin^2(\kappa \arcsin\sqrt{x_n}). \tag{3.11}$$

В отличие от систем, рассматриваемых ранее, преобразование $x_{n+1} = f(x_n)$ здесь может быть *неоднозначным*, в частности, когда κ — нерациональное число. Неоднозначность предполагает, что заданному значению x_n может соответствовать не одно, а множество возможных последующих состояний x_{n+1}. На рис. 3.15 представлен пример неоднозначного преобразования (3.10) при $\kappa = \pi^{1/3}$.

Хаотические системы с точным решением и неоднозначным преобразованием представляют большой интерес для криптографии. С одной стороны, точное решение позволяет получить доступ к произвольному члену последовательности, однозначно определяемой начальным условием (семенем). С другой стороны, значения x_{n-1} и x_{n+1} не могут быть вычислены из x_n, то есть генератор может быть непредсказуемым в один шаг.

Функция $x_n = \Psi(\theta, \kappa, n)$ должно быть однонаправленным, то есть вычисление начальных условий x_0 из x_n должно быть практически невозможным.

Уравнение 3.10 относительно θ и κ при известном x_n имеет бесчисленное множество решений. Преобразование x_n в символ в соответствии с разбиением (X_0, X_1) тоже может является однонаправленной, так как связана с большой потерей информации. Однако, непредсказуемость семени, при известной символьной траектории большой длины требует дополнительного исследования.

Вычисление точного решения часто содержит операцию «растягивание—сжатие». Сначала аргумент растягивается на некоторый большой отрезок (умножением, возведением в степень), а затем сжимается в исходное пространство состояний (при помощи периодической функции, например, $\sin$, mod). Это приводит к тому, что длина последовательности, которую можно получить на машине с конечной точностью вычислений, тоже ограничена. Таким образом, хаотические системы с точным решением тоже имеют тоже свой «горизонт» предсказуемости. Более того, при больших n в результате значительного округления промежуточных результатов, последовательность может существенно отличаться от хаотической и содержать заметные паттерны.

Котульский *et al.* [61] провел ряд тестов на псевдослучайность для некоторых известных хаотических систем с точным решением и неоднозначным преобразованием (в частности, (3.10)). Точное решение не позволило построить программный генератор псевдослучайных чисел, отвечающий требованиями современной криптографии. В частности, статистические свойства последовательности сильно менялись в зависимости от начальных условий. Вопрос применения псевдохаотических системы с точным решением в криптографии требует дополнительного исследования.

3.3 Бинарный псевдохаос

3.3.1 Общие свойства

В разделе 1.2.6 мы дали определение дискретной хаотической системы, определенной на множестве бесконечных бинарных строк Ω. Преобразование такой системы задано бинарной функцией $f : \Omega \to \Omega$.

Криптографические системы построены в пространстве состояний из конечных строк и, согласно нашей терминологии, называются псевдохаотическими. Бинарные псевдохаотические системы на начальных итерациях проявляют экспоненциальную неустойчивость, но при дальнейшем течении дискретного времени становится заметным периодическое поведение.

Показатель экспоненты Ляпунова для бинарной псевдохаотической системы $\langle \mathcal{A}, f \rangle$, $\mathcal{A} \subset \left\{ \alpha_m | \alpha \in \{0,1\}^m \right\}$ с конечным число состояний 2^m определяется выражением

$$\lambda(\alpha_0) = \lim_{n\to\infty} \frac{1}{n} \log d_H\left(f^n(\alpha_0), f^n\left(\alpha_0'\right)\right)$$

где $\alpha_0' \in \mathcal{A}$ такое, что $d_H(\alpha_0, \alpha_0') = 1$.

В «хорошей» псевдохаотической системе с n-битовым состоянием, соседние траектории (с началом в x_0 и x_0', такие что $d_H(x_0, x_0') = 1$) экспоненциально расходятся, в среднем, до $\langle d_H \rangle = m/2$ (рис. 3.3-b).

Как всякая динамическая система с конечным числом состояний, псевдохаотическая система является периодической. Генератор чисел, в котором существует всего одна орбита, (ее длина равна числу элементов пространства состояний), называют идеальным. Например, генератор с линейной обратной связью (LFSR)

$$x_n = (c_1 x_{n-1} + c_2 a_{n-2} + \cdots + c_k a_{n-k}) \bmod 2$$

при определенных значениях коэффициентов c_i является идеальным [75].

Многие бинарные псевдохаотические системы имеют орбиты, длина которых зависит от начальных условий. Так, Фог (Fog, [40]) исследовал так называемое преобразование RANROT. Это преобразование отличается от обыкновенного аддитивного генератора тем, что имеет дополнительный сдвиг:

$$x_n = \left((x_{n-j} + x_{n-k}) \bmod 2^b\right) \operatorname{rotr} r,$$

где $x_n \in \{0,1\}^b, r \in [0, b/2)$ и числа j, k выбираются в соответствии с некоторым правилами. Генератор обладает хорошими статистическим свойствами и является вычислительно непредсказуемым при неизвестных параметрах. Как уже было отмечено, переменная длина орбиты опасна тем, что существует вероятность появления паттернов. Фог считает генератор крайне эффективным и рекомендует проводить тест орбиты (до 1000 итераций) перед началом работы криптографического приложения.

3.3.2 Дискретное палаточное преобразование

Масуда *et al.* (Masuda, [67]) разработали криптосистему на базе дискретной версии одномерного палаточного преобразования (рис. 3.16)

$$F(X) = \begin{cases} \left\lceil \frac{M}{A} X \right\rceil, & 1 \le X \le A \\ \left\lfloor \frac{M}{M-A}(M-X) + 1 \right\rfloor, & A < X \le M \end{cases},$$

где $X, A = 1, 2, ..., M$ и $\lfloor\,\rfloor, \lceil\,\rceil$ — округление в меньшую и большую строну соответственно. Начальное условие криптосистемы задается открытым текстом, а заключительное состояние определяет шифротекст. Ключем является параметр

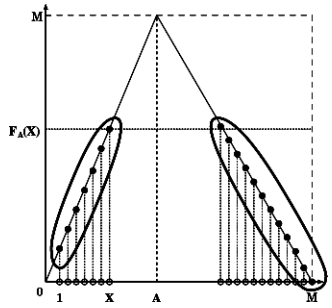

Рис. 3.16. Дискретное палаточное преобразование.

A. Преобразование применяется N раз. Автор находит минимальное значение N, при котором система достаточно устойчива против линейного и дифференциального криптоанализа.

3.3.3 Клеточные автоматы

Вульфрам (Wolfram, [86]) впервые применил одномерный клеточный автомат для генерации псевдослучайных чисел. Состояние системы определяется массивом клеток (битов)

$$\mathbf{b} = (b\,(1)\,, b\,(2)\,, \ldots, b\,(n)) \in \{0,1\}^n\,.$$

Итерационная функция $f : \{0,1\}^n \to \{0,1\}^n$ задана выражением

$$b(i) = b(i) - 1\, xor\, \big(b(i)\, or\, b(i+1)\big),$$

для всех $1 \leq i \leq n$. Массив бит считается циклическим, то есть $b\,(n+1) = b\,(1)$. Все элементы обновляются параллельно, а на выход выдается только один бит $b_k, k \in [0,n]$.

Риттер (Ritter, [75]) отмечает, что последовательность такого генератора является псевдослучайной, однако, длина периода непостоянна и непредсказуема. При относительно простой аппаратной реализации, генератор не является эффективным в программном исполнении на одном процессоре (особенно при больших размерах массива).

Гутовитз (Gutowitz, [48]) разработал сложную блочную схему шифрования на базе клеточных автоматов. Блок открытого текста (512 бит) преобразуется в блок шифротекста несколько большего размера (578 бит). Ключ (1088 бит), состоящий из двух частей использует для особой схемы шифрования, так называемой структуры блок-связь. Используется две итерации, каждая из которых

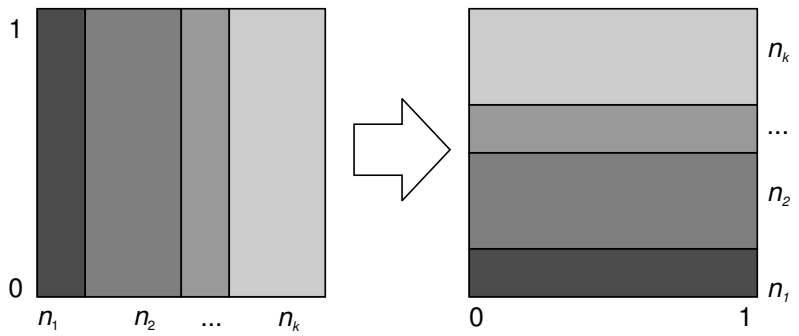

Рис. 3.17. Обобщенное преобразование пекаря.

включает две подитерации. В свою очередь, каждая подитерация включает фазы подстановки (s-box) и перемешивания (p-box). Автор считает, что схема шифрования обеспечивает псевдослучайный вид шифротекста, а так же является устойчивой к дифференциальному криптоанализу.

3.3.4 Обобщенное преобразование пекаря

Растягивающее и сжимающее преобразование пекаря было упомянуто применительно к криптографии еще Шенноном [79]. Мы рассмотрим двумерную систему на базе обобщенного преобразования пекаря, так же называемую потоками Колмогорова.

Квадратный кусок теста разрезается на вертикальные полосы в соответствии с разбиением $\rho = \{n_1, n_2, ..., n_k\}$. Каждая полоса по горизонтали растягивается до длины стороны квадрата, а по вертикали сжимается до исходной ширины n_i, как показано на рис. 3.17.

Система на базе преобразования пекаря проявляет экспоненциальную чувствительность по отношению к начальным условиям и параметру (разбиению ρ). Так, всего 6 итераций, перемешивают картинку до неузнаваемости (рис. 3.18).

Дискретное преобразования пекаря может быть реализована при помощи модульной арифметики. Тогда итерационная функция T_ρ задана на матрице из N элементов:

$$T_\rho : \{0,1\}^N \to \{0,1\}^N$$

Набор целых чисел $\{n_1, n_2, ..., n_k\}$ выбирается так, что каждое число n_i делит без остатка N и суммма $n_1 + n_2 + ... + n_k = N$. Преобразование T_ρ задает новую

Рис. 3.18. Шесть итераций преобразования пекаря с разбиением $\rho = \{0.25, 0.5, 0.25\}$, примененных к изображению [76].

позицию элемента матрицы с координатами (r, s):

$$T_\rho(r,s) = \left(\frac{N}{n_i}(r - N_i) + s \bmod \frac{N}{n_i}, \frac{n_i}{N}\left(s - s \bmod \frac{N}{n_i} \right) + N_i \right),$$

где $r, s \in \{1, 2, ..., N\}, N_i = n_1 + n_2 + ... + n_i, i \in \{1, 2, ..., k\}$ и $N_i \leq r < N_i + n_i$. Преобразование T_ρ осуществляет только *циклическую* перестановку, но не меняет статистические свойства открытого текста. Таким образом, в блочной схеме шифрования преобразование пекаря можно использовать для организации фазы перестановки (p-box), а для фазы подстановки (s-box) необходимо дополнительное преобразование, обеспечивающее запутывание.

Шарингер (Scharinger) [76] и Фридрих (Fridrich) [42] разработали блочную схему шифрования на базе шифрования пекаря. Исследователи считают, что псевдохаотическая система позволяет эффективно перемешивать блоки большого размера, например, цифровое изображение. С помощью простого модульного сложения, Фридрих расширил двумерное преобразовавшие пекаря до трехмерного. Это комбинированное преобразование обеспечило псевдослучайное распределение шифротекста уже после нескольких итераций. Распыление (то есть установление взаимосвязи между каждым битом открытого текста и ключа со *всей* последовательностью шифротекста) осуществляется при помощи обычного регистра с линейной обратной связью (LFSR) [75]. Каппеллетти (Cappelletti, [31]) разработал аппаратную реализацию схемы шифрования Шарингера на базе программируемой микросхемы.

3.3.5 Псевдохаос в классических криптосистемах

Стандартные криптографические системы представляют собой бинарный псевдохаос, то есть системы с ограниченной «экспонентой Ляпунова» и топологической транзитивностью на множестве битовых строк.

Псевдослучайные генераторы

В разделе 2.6 мы уже увидели, что обычный линейный конгруэнтный генератор псевдослучайных чисел является дискретным аналогом хаотической системы на базе пилообразного преобразования. Однако, помимо чувствительности к начальным условиям, криптографическая стойкость генератора определяется еще одним важным параметром — вычислительной непредсказуемостью преобразования $x_{n+1} = f(x_n)$. В классической криптографии эта непредсказуемость достигается за счет неразрешимости некоторой задачи в теории чисел.

Например, генератор Blum-Blum-Shub [69, 75] построен на проблеме разложения большого числа на множители. Итерационная функция задана выражением

$$x_{n+1} = x_n^2 \bmod M,$$

где $M = pq$, p, q — простые числа, конгруэнтные $3 \bmod 4$. Выходной бит генератора $b_n = \sigma(x_n)$, где $\sigma(x_n)$ — однонаправленная функция, взятие последнего бита x_n.

Неразрешимые задачи могут быть найдены и в нелинейной динамике:

1) Сильное растягивающее и сжимающее преобразование, чувствительное к управляющему параметру.
2) Композиция некоторого числа хаотических, смешивающий преобразований поражающих асимптотически случайную последовательность (промежуточные состояния недоступны для внешнего наблюдателя).

Симметричные блочные шифры

Обычные блочные шифры представляют собой совокупность нескольких хаотических, смешивающий преобразований. Проиллюстрируем это на примере схемы шифрования Райндол (Rijndael, [73])[2]. Райндол является симметричным, итерационном, блочном шифром. Состояние криптосистемы задано двумерным массивом (матрицей) битов. Начальное состояние системы является открытом текстом. Комбинированная итерационная функция шифра задана:

[2]Алгоритм Райндол в 2001 году принят в качестве стандарта AES (Advanced Encryption Standard), оговаривающий шифрование важных, неклассифицированных документов правительства США. Вероятно, станет стандартом де-факто и в коммерческом секторе.

1) фазой подстановки (s-box), реализованной при помощи обратного мультипликативного и аффинного преобразований (последнее, кстати, хорошо известно в теории хаоса и фракталов);
2) фазой перемешивания (p-box), включающей циклический сдвиг строк и перестановку столбцов;
3) фазой сложения с итерационном ключем (round key). То есть к состоянию системы добавляется псевдослучайная матрица, которая генерируется при помощи отдельной псевдохаотической подсистемы.

Эта комбинированная итерационная функция применяется к состоянию системы $N > 10$ раз. «Полное запутывание» (каждый бит открытого текста и ключа влияет на каждый бит шифротекста) достигается уже после двух итераций. Заключительное состояние является шифротекстом.

Глава 4

Заключение

4.1 Теоретические выводы

1) Существует фундаментальная взаимосвязь между криптографическими и хаотическими системами. И там и здесь осуществляется итерационное преобразование информации в соответствии с некоторым детерминированным законом. Чувствительность к начальным условиям и смешивание, в своей совокупности, реализуют запутывание и распыление — два криптографических метода, предложенные Шенноном.
2) В то же время существует фундаментальные различие между криптографией и теорией хаоса: (1) криптография изучает системы с конечным числом состояний, а теория хаоса — бесконечные пространства; (2) криптография изучает результат конечного числа итераций ($n < \infty$), а теория хаоса — асимптотическое поведение системы ($n \to \infty$); (3) криптография изучает вычислительно непредсказуемые системы, а теория хаоса — системы разной степени предсказуемости.
3) Хаотические системы являются алгоритмически случайным и не могут быть точно предсказаны компьютером даже с бесконечной вычислительной мощностью. С другой стороны, хаотические системы могут быть предсказаны вероятностной машиной.
4) Выборка $x_k, x_{2k}, \ldots x_{nk}, \ldots$ из последовательности $x_1, x_2, x_3, \ldots$, полученной при помощи хаотического и смешивающего преобразования может быть асимптотически случайной, то есть, с увеличением k элементы $x_{(n-1)k}$ и x_{nk} будут все более независимы.
5) Хаотические системы с точным решением $x_n = \Psi(x_0, n)$ и неоднозначным преобразованием $x_{n+1} = f(x_n)$ позволяют получить вычислительно непредсказуемые последовательности. Преимуществом генератора на ба-

зе такой системы, является то, что произвольный элемент x_n может быть быстро вычислен независимо от предшествующих. Секретность всей последовательности заложена в начальных условиях x_0 и преобразовании f.

4.2 Практические выводы

1) Применение хаотических систем, аппроксимированных при помощи математики с плавающей запятой (МПЗ), в криптографии на сегодняшний момент не оправдано. Такие системы могут неожиданно войти в легко предсказуемый режим работы (например, в колебательный с коротким периодом или даже в состояние покоя). Пока не существует эффективных методов предотвращения коротких орбит и соответствующих им паттернов. Хаотические системы, построенные на базе элементарных нелинейных функций (например, x^2, $\sin(x)$, $\log(x)$) часто являются вычислительно предсказуемыми, то есть имеют закон распределения, легко отличимый от равномерного.
2) Все современные криптографические системы (схемы шифрования, псевдослучайные генераторы, хеш-функции) можно считать бинарными псевдохаотическими системами. Такие системы задаются бинарной итерационной функцией (или системой функций) на конечном множестве состояний (битовых строк). Криптосистемы обладают ограниченной экспоненциальной неустойчивостью (чувствительностью к начальным условиям) и топологической транзитивностью. Дополнительно свойство — смешивания — обуславливает псевдослучайность последовательности.
3) Бинарные псевдохаотические системы могут иметь (a) одну или (b) множество орбит. В последнем случае, возможны варианты, когда все орбиты имеют (b1) одинаковую или (b2) разную длину. Определение длины минимальной орбиты (в общем случае, нахождение функции распределения длин орбит) — задача нетривиальная и требует частого решения для каждой нелинейной системы.
4) Блочные схемы шифрования (например, DES, AES) часто представляют собой совокупность некоторых псевдохаотических преобразований. Их суммарным эффектом является более сложное и непредсказуемое, псевдохаотическое преобразование со смешиванием, которые обеспечивает шенноновское запутывание и распыление.
5) Обычные псевдослучайные генераторы (например, RSA) опираются на неразрешенную проблему из теории чисел (разложение большего числа на простые множители). На основе проделанной работы мы можем пред-

ложить, по сути, близкие подходы к построению генераторов:

На базе псевдохаотической смешивающей системы с точным решением и неоднозначным преобразованием. С точки зрения использования, такой генератор удобнее, так как позволяет получить произвольный элемент последовательности *без* вычисления предшествующих. Решение системы $x_n = \Psi(x_0, n)$ должно быть однонаправленным, то есть вычисление семени (начальных условий x_0 и ключа) из последовательности $x_1, x_2, \ldots$ должны быть практически не осуществимом.

На базе псевдохаотической смешивающей системы, в которой k-кратное преобразование $x_k = f^k(x_0)$ вычислительно непредсказуемо при неизвестном параметре (ключе) и достаточно большом k. Тогда, скрывая от наблюдателя промежуточные состояния $x_{nk+1}, x_{nk+2}, \ldots, x_{nk+(k-1)}$, генератор будет выдавать асимптотически случайную последовательность $x_k, x_{2k}, \cdots, x_{kn}, \cdots$

6) Технически, псевдослучайные генераторы построены на известном механизме хаотического преобразования — «растягивание—сжатие». Сначала состояние системы растягивается на некоторое большое пространство (умножением, возведением в степень), а потом сжимается в исходное пространство состояний (при помощи периодической функции mod). Это свойство хорошо знакомо в теории хаоса, например, в системах на базе пилообразного преобразования или преобразования пекаря.
7) Растягивание и сжатие происходит и в случае точного решения Сильное растягивание и сжатие в МПЗ-аппроксимированных приводит к тому, что, несмотря на отсутствие накопительной ошибки, последовательность качественно отличается от хаотической и повторяться, причем длина периода непредсказуема и зависит от начальных условий.

4.3 Дальнейшая работа

Интересными направлениями для дальнейших исследований являются:

1) Поиск и оценка криптографических свойств бинарных псевдохаотических систем с точным решением и псевдослучайным вероятностным ансамблем;
2) Поиск условий вычислительной непредсказуемости (псевдослучайности) хаотических и псевдохаотических систем;
3) Поиск условий, при которых хаотическая и псевдохаотическая системы являются устойчивыми к дифференциальному и линейному криптоанализам;
4) Исследование бинарного псевдохаоса, в которых в процессе генерации по-

следовательности увеличивается длина битовой строки состояния.

5) Разработка блочных схем шифрования на базе бинарных псевдохаотических систем с сильным смешиванием и сложным нелинейным преобразованием.

Литература

[1] А. Ю. Лоскутов, *Синергетика и нелинейная динамика: новые подходы к старым проблемам*, Синергетика. Труды семинара. Том 3. М.: Изд-во МГУ. 2000.

[2] С. П. Капица, С. П. Курдюмов, and Малинецкий Г. Г., *Синергетика и прогнозы будущего*, Едиториал УРСС, 2001.

[3] М. В. Капранов and В. Г. Чернобаев, *Управляемые генераторы хаотических колебаний на базе систем фазовой синхронизации*, Радиотехнические тетради (1998), no. 15.

[4] Г. Г. Малинецкий and А. Б. Потапов, *Современные проблемы нелинейной динамики*, Едиториал УРСС, 2000.

[5] М. Гэри and Д. Джонсон, *Вычислительные машины и трудно решаемые задачи*, Мир, Москва, 1982.

[6] Г. Николис and И. Пригожин, *Познание сложного. Введение.*, Мир, 1990, // Exploring complexity. An introduction. G. Nicolis, and I. Prigogine. W. H. Freeman and Company, New York.

[7] А. Дмитриев and С. Старков, *Новые подходы к решению проблем в системах связи и компьютерных сетях: динамический хаос*, Компьютерра (2001), no. 46.

[8] П. Р. Халмош, *Теория меры*, ИЛ, 1953.

[9] Я. Г. Синай, *Современные проблемы эргодической теории*, ФИЗМАТЛИТ, Москва, 1995.

[10] В. С. Анищенко, *Детерминированный хаос*, Соросовский образовательный журнал (1997), no. 6, 70–76, `http://www.nsu.ru/materials/ssl/text/metodics/anishenko.html`.

[11] Дж. Д. Биркгоф, *Динамические системы*, РХД, Ижевск, 1999.

[12] П. Р. Халмош, *Лекции по эргодической теории*, РХД, Ижевск, 1999.

[13] В. В. Суриков, *О термине синергетика*, Синергетика. Труды семинара. (Москва), vol. 3, Изд-во МГУ, 2000.

[14] В. В. Ященко (ed.), *Введение в криптографию*, МЦНМО, Москва, 2000.

[15] А. Ю. Лоскутов, *Синергетика и нелинейная динамика: новые подходы к*

старым проблемам, Синергетика. Труды семинара. (Москва), vol. 3, Изд-во МГУ, 2000.

[16] Б. К. Мартыненко, *Языки трансляции*, Издательство С.-Петербургского Университета, 2001.

[17] Г. Шустер, *Детерминированный хаос. Введение*, Мир, Москва, 1988, // H. G. Schuster, Deterministic chaos: an introduction, VCH, Weinheim, 1988.

[18] А. Лоскутов, *Нелинейная динамика, теория динамического хаоса и синергетика (перспективы и приложения)*, Компьютерра (1998), no. 47, `http://www.cplire.ru/win/InformChaosLab/chaoscomputerra/Loskutov.html`.

[19] И. Пригожин, *Конец определенности. Время, хаос и новые законы*, РХД, Ижевск, 1999, // I. Prigogine, The end of certainty. Time, chaos and the new laws of nature, The Free Press, New York, 1997.

[20] А. Дмитриев, *Детерминированный хаос и информационные технологии*, Наука и жизнь (2001), no. 5, `http://nauka.relis.ru/cgi/nauka.pl?07+0105+07105044+HTML`.

[21] М. Шредер, *Фракталы, хаос, степенные законы. Миниатюры из бесконечного рая*, РХД, Ижевск, 2001, // M. Schroeder, Fractals, Chaos, Power Laws. Mintutes from an infinite paradise, W. H. Freeman and Company, New York.

[22] *Большая советская энициклопедия*, Советская энциклопедия, 1969–1978, `http://www.rubicon.ru/`.

[23] *A statistical test suite for the validation of random number generators and pseudo random number generators for cryptographic applications*, NIST, 2001, `http://csrc.nist.gov/rng/rng2.html`.

[24] А. Н. Колмогоров, *Теория информации и теория алгоритмов*, Наука, Москва, 1987.

[25] M. S. Baptista, *Cryptography with chaos*, Physics Letters A **240** (1998), no. 1–2, 50–54.

[26] E. Beham, *Cryptanalysis of the chaotic-map cryptosystem*, Proc. of EUROCRYPT'91, 1991, `http://citeseer.nj.nec.com/175190.html`.

[27] M. E. Bianco and D. Reed, *An encryption system based on chaos theory*, US Patent No. 5048086, 1991.

[28] M. Blum and S. Micali, *How to generate crytographically strong sequences of pseudo-random bits*, SIAM Journal of Computations **13** (1984), no. 4, 850–864.

[29] G. Boffetta, M. Cencini, M. Falcioni, and A. Vulpiani, *Predictability: a way to characterize complexity*, 2001, `http://www.unifr.ch/econophysics/`.

[30] R. Brown and L. O. Chua, *Clarifying chaos: examples and counterexamples*, IJBC **6** (1996), no. 2, 219–249.

[31] L. Cappelletti, *An fpga implementation of a chaotic encryption algorithm*, Master's thesis, Universita Degli Studi di Padova, 2000, `http://www.lcappelletti.`

f2s.com/Didattica/thesis.pdf.

[32] J. M. Carroll, J. Verhagen, and P. T. Wong, *Chaos in cryptography: the escape from the strange attractor*, Cryptologia **16** (1992), no. 1, 52–72.

[33] Y. H. Chu and S. Chang, *Dynamic cryptography based on synchronized chaotic systems*, Electronic Letters **35** (1999), no. 12.

[34] _____, *Dynamic data encryption system based on synchronized chaotic systems*, Electronic Letters **35** (1999), no. 4.

[35] F. Dachselt, K. Kelber, and W. Schwarz, *Chaotic coding and cryptoanalysis*, citeseer.nj.nec.com/355232.html.

[36] F. Dachselt, K. Kelber, J. Vandewalle, and W. Schwarz, *Chaotic versus classical stream ciphers – a comparative study*, 1998, citeseer.nj.nec.com/article/dachselt98chaotic.html.

[37] W. Ebeling, L. Molgedey, J. Kurths, and U. Schwarz, *Entropy, complexity, predictability and data analysis of time series and letter sequences*, 1999, http://citeseer.nj.nec.com/395066.html.

[38] T. Erber, T. Rynne, W. Darsow, and M. Frank, *The simulation of random processes on digital computers: unavoidable order*, Journal of Computational Physics (1983), no. 49, 349–419.

[39] M. J. Feigenbaum, *The universal metric properties of nonlinear transformations*, J. Stat. Physics (1979), no. 21, 669–706.

[40] A. Fog, *Chaotic random number generators*, 1999, http://www.agner.org/random/theory/.

[41] J. Fridrich, *Discrete-time dynamical systems under observational uncertainty*, J. Appl. Math. Comp. (1993), no. 83, 181–207.

[42] _____, *Symmetric ciphers based on two dimension chaotic map*, IJBC **8** (1998), no. 6, 1259–1284.

[43] J. Fridrich and J. Geer, *Reconstruction of chaotic orbits under finite resolution*, J. Appl. Math. Comp. (1995), no. 80, 129–159.

[44] P. Gacs, *Lecture notes on descriptional complexity and randomness*, Computer Science Department, Boston University, 2001, http://cs-pub.bu.edu/faculty/gacs/Home.html.

[45] J. B. Gallagher and J. Goldstein, *Sensitive dependence cryptography*, 1996, http://www.navigo.com/sdc/.

[46] O. Goldreich, *Introduction to complexity theory*, Lecture Note, Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Israel, 1999.

[47] J. A. González and R. Pino, *Chaotic and stochastic functions*, Physica **276A** (2000), 425–440.

[48] H. Gutowitz, *Cryptography with dynamical systems*, ESPCI, Laboratoire

d'Electronique, Paris, France, 1995, http://www.santafe.edu/~hag/crypto/crypto.html.

[49] T. Habutsu, Y. Nishio, I. Sasase, and S. Mori, *A secret key cryptosystem by iterating chaotic map*, Proc. of EUROCRYPT'91, 1991, http://link.springer.de/link/service/series/0558/bibs/0547/05470127.htm, pp. 127–140.

[50] J. Hastad, *Pseudo-random generators under uniform assumptions*, Proceedings 22nd Annu. ACM Symp. on Theory of Computing, 1990, pp. 385–404.

[51] M. K. Ho, *Chaotic encryption techniques*, Master's thesis, Department of Electronic Engineering, City University of Hong Kong, 2001, http://personal.cityu.edu.hk/~50115849/ces/.

[52] S. Hollasch, *Ieee standard 754: floating point numbers*, 1998, http://research.microsoft.com/~hollasch/cgindex/coding/ieeefloat.html.

[53] J. Hosack, *The use of chebysev mixing to generate pseudo-random numbers*, Journal of Computational Physics (1986), no. 67, 482–486.

[54] R. Impagliazzo, L. Levin, and M. Luby, *Pseudo-random generation from one-way functions*, Proc. 21st Annu. ACM Symp. on Theory of Computing, 1989, pp. 230–235.

[55] E. A. Jackson, *Perspectives in nonlinear dynamics*, Cambridge University Press, 1991.

[56] M. P. Kennedy, R. Rovatti, and G. Setti, *Chaotic electronics in telecommunications*, CRC Press, 2001.

[57] D. Knuth, *The art of computer programming - seminumerical algorithms*, vol. 2, 2nd ed. Addison-Wesley: Reading, Massachusetts, 1981.

[58] L. Kocarev, *Chaos and cryptography*, 2001, http://rfic.ucsd.edu/chaos/ws2001/kocarev.pdf.

[59] L. Kocarev, *Chaos-based cryptography: a brief overview*, Circuits and systems **1** (2001), no. 3, 6–21.

[60] L. J. Kocarev, K. S. Halle, K. Eckert, and L. O. Chua, *Experimental demonstration of secure communications via chaotic synchronization*, IJBC **2** (1992), no. 3, 709–713.

[61] Z. Kotulski, J. Szczepańs, K. Gyrski, A. Gyrska, and A. Paszkiewicz, *On constructive approach to chaotic pseudorandom number generators*, Proc. of the Regional Conference on Military Communication and Information Systems, vol. 1, CIS Solutions for an Enlarged NATO, RCMIS, 2000, http://www.ippt.gov.pl/~zkotulsk/CPRBG.pdf, pp. 191–203.

[62] Z. Kotulski and J. Szczepański, *Discrete chaotic cryptography. new method for secure communication*, Proc. NEEDS'97, 1997, http://www.ippt.gov.pl/~zkotulsk/kreta.pdf.

[63] Z. Kotulski and J. Zczepanski, *On the application of discrete chaotic dynamical*

systems to cryptography. dcc method, Biuletyn Wat Rok **XLVIII** (1999), no. 10, 111–123, http://www.ippt.gov.pl/~zkotulsk/wat.pdf.

[64] D. Lai, G. Chen, and M. Hasler, *Distribution of the lyapunov exponent of the chaotic skew tent map*, IJBC **9** (1999), no. 10, 2059–2067.

[65] L. A. Levin, *One-way function and pseudorandom generators*, Combinatorica **7** (1987), no. 5, 357–363.

[66] L. Lovász, *Computation complexity*, Lecture Notes, http://ftp.cs.yale.edu/pub/lovasz.pub/.

[67] N. Masuda and K. Aihara, *Finite state chaotic encryption system*, 2000, http://www.aihara.co.jp/rdteam/fs-ces/.

[68] R. Matthews, *On the derivation of a chaotic encryption algorithm*, Cryptologia (1989), no. 13, 29–42.

[69] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of applied cryptology*, CRC Press, 1996, http://www.cacr.math.uwaterloo.ca/hac/.

[70] V. I. Oseledec, *A multiplicative ergodic theorem: Lyapunov charcteristic numbers for dynamical systems*, Trans. Mosc. Math. Soc. **19** (1968), no. 197.

[71] N. Paar, *Robust encryption of data by using nonlinear systems*, 1999, http://www.physik.tu-muenchen.de/~npaar/encript.html.

[72] V. A. Protopopescu, R. T. Santoro, and J. S. Tolliver, *Fast and secure encryption-decryption method*, US Patent No. 5479513, 1995.

[73] V. Rijmen and J. Daemen, *Rijndael algorithm specification*, 1999, http://www.esat.kuleuven.ac.be/~rijmen/rijndael/.

[74] T. Ritter, *Ciphers by ritter*, 1991, http://www.ciphersbyritter.com/.

[75] ———, *The efficient generation of cryptographic confusion sequences*, Cryptologia (1991), no. 15, 81–139, http://www.ciphersbyritter.com/ARTS/CRNG2ART.HTM.

[76] J. Scharinger, *Secure and fast encryption using chaotic kolmogorov flows*, Johannes Kepler University, Department of System Theory, 1998, http://www.cast.uni-linz.ac.at/Department/Publications/Pubs1998/Scharinger98f.htm.

[77] B. Schneier, *Applied cryptography, second edition*, John Wiley & Sons, Inc, 1996, ISBN 0-471-12845-7, http://www.ssl.stu.neva.ru/psw/crypto/appl_rus/appl_cryp.htm.

[78] C. E. Shannon, *A mathematical theory of communication*, Bell System Technical Journal **27** (1948), no. 4, 379–423, 623–526.

[79] ———, *Communication theory of secrecy systems*, Bell System Technical Journal **28** (1949), no. 4, 656–715.

[80] M. I. Sobhy and A. E. D. Schehata, *Secure e-mail and databases using chaotic encryption*, Electronic Letters **36** (2000), no. 10.

[81] M. Svensson and J. E. Malmquist, *A simple secure communications system utilizing chaotic functions to control the encryption and decryption of messages*, Project report for the course 'Chaos in science and technology', Lund Institute of Technology, Dept. of physics, Subdept. of mathematical physics, 1996, `http://www.efd.lth.se/~d92ms/chaoscrypt.html`.

[82] J. Szczepański and Z. Kotulski, *Pseudorandom number generators based on chaotic dynamical systems*, Open Systems & Information Dynamics **8** (2001), no. 2, 137–146, `http://www.ippt.gov.pl/~zkotulsk/open.pdf`.

[83] H. Waelbroeck and F. Zertuche, *Discrete chaos*, 1998, `http://papaya.nuclecu.unam.mx/~nncp/chaos98.ps`.

[84] D. D. Wheeler, *Problems with chaotic cryptosystems*, Cryptologia (1989), no. 12, 243–250.

[85] ———, *Supercomputer investigations of a chaotic encryption algorithm*, Cryptologia (1991), no. 15, 140–150.

[86] S. Wolfram, *Random sequence generation by cellular automata*, Advances in Applied Mathematics (1986), no. 7.

[87] W. K. Wong, *Chaotic encryption technique*, City University of Hong Kong, Department of Electronic Engineering, Hong Kong, 1999, `http://kitson.netfirms.com/chaos/`.

[88] C. J. Wu and Y. C. Lee, *Observer-based method for secure communication of chaotic systems*, Electronic Letters **36** (1999), no. 22.

[89] A. C. Yao, *Theory of applications of trapdoor functions*, Proc. of IEEE Symp. on Foundations of Computer Science, 1982, pp. 80–91.

Предметный указатель