РОССИЙСКАЯ АКАДЕМИЯ НАУК
СИБИРСКОЕ ОТДЕЛЕНИЕ
ИНСТИТУТ ВЫЧИСЛИТЕЛЬНЫХ ТЕХНОЛОГИЙ

На правах рукописи

СТОГНИЕНКО Владимир Сергеевич

РАЗРАБОТКА И ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ АЛГОРИТМОВ ТЕСТИРОВАНИЯ СЛУЧАЙНЫХ ЧИСЕЛ И ИХ ПРИЛОЖЕНИЯ К НЕКОТОРЫМ ЗАДАЧАМ КРИПТОГРАФИИ

05.13.18 – математическое моделирование, численные методы и комплексы программ

А В Т О Р Е Ф Е Р А Т

диссертации на соискание ученой степени
кандидата технических наук

Новосибирск - 2004

Работа выполнена в Институте вычислительных технологий Сибирского отделения Российской академии наук

Научный руководитель: доктор технических наук,
профессор Б. Я. Рябко
Научный консультант: академик РАН
Ю. И. Шокин

Официальные оппоненты: доктор технических наук,
профессор Елипов Б.С.

кандидат технических наук,
доцент Фионов А.Н.

Ведущая организация: Институт вычислительного моделирования СО РАН
(г. Красноярск).

Защита состоится "8" сентября 2004 г. в 16.00 на заседании диссертационного совета Д 003.046.01 при Институте вычислительных технологий СО РАН по адресу: 630090 Новосибирск, проспект Академика Лаврентьева, 6

С диссертацией можно ознакомиться в специализированном читальном зале вычислительной математики и информатики отделения ГПНТБ.

Автореферат разослан "_2_" августа 2004 г.

Ученый секретарь
диссертационного совета
доктор физико-математических наук Чубаров Л.Б.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы*. Случайные и псевдослучайные числа широко используются в различных областях науки и техники. Они очень важны в математическом моделировании, в разработке численных методов, в программировании, криптографии и т. п. И если раньше ученые, нуждающиеся для своей работы в случайных числах, раскладывали карты, бросали кости или вытаскивали шары из урны, которую предварительно "как следует трясли", то сейчас генератор случайных чисел встроен в любой компилятор и процессор компьютера. Например, Intel использует генератор случайных чисел в процессорах со второго полугодия 1999 года. Однако, такие генераторы, как правило, недостаточно надежны для криптографических приложений. На создание удовлетворительных псевдослучайных последовательностей с помощью численных методов затрачена масса усилий. В литературе можно найти множество работ по генераторам псевдослучайных чисел, а также различные тесты на случайность. Тем не менее, проблема получения псевдослучайных чисел и их тестирования все еще актуальна и поэтому находится в центре внимания многих исследователей.

В научно-технической литературе числа, полученные с помощью компьютера, называются псевдослучайными, мы же, в дальнейшем для краткости, будем называть их просто случайными или случайными последовательностями. Каждую последовательность случайных чисел, перед использованием, необходимо тщательно проверить. Для приложений и задач, в которых требуются действительно "качественные" случайные числа, они проверяются специальным набором тестов. Формально генератор псевдослучайных чисел можно считать эффективным, если он проходит все известные статистические тесты. В связи с этим, статистические тесты важны и необходимы, также как и сами генераторы. Более того, генератор случайных чисел может считаться "идеальным" только до тех пор, пока не появится тест, доказывающий обратное и "время жизни" генератора может на этом остановиться.

Необходимо отметить что, по сравнению с другими задачами, криптографические приложения предъявляют к генераторам случайных последовательностей намного более строгие требования, чем, скажем, вычислительные задачи. "Криптографическая" случайность – это не просто статистическая случайность, хотя и включает ее. Чтобы псевдослучайная последовательность была криптографически стойкой, она должна обладать следующими свойствами: генерируемые последовательности проходят все статистические тесты и их символы должны быть непредсказуемы. В то же время, как и любой криптографический алгоритм, генераторы криптографически стойких случайных последовательностей служат объектами "вскрытия" или "взлома". Поэтому, экспериментальные исследования генераторов криптографически стойких случайных последовательностей, разработка эффективных алгоритмов и методов их тестирования, их устойчивость к "взлому", т. е. проверка качества получаемых последовательностей – важная задача криптографии.

Среди задач тестирования последовательностей "на случайность" особый интерес представляет задача различения статистическими методами зашифрованных текстов и случайных последовательностей. Разработка эффективных алгоритмов и методов распознавания, или различения зашифрованных текстов довольно хорошо известно в

* Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (коды проектов: 99-01-00586, 03-01-00495 и INTAS-00-738 "Эффективное кодирование источников информации и связанные проблемы".

криптографии (см., например, монографию Б. Шнайера “Прикладная криптография”, М., 2002 г.). Решение этой задачи позволяет определить сам факт передачи по сетям связи (или хранения в информационных базах данных) зашифрованных текстов и, в силу этого, представляет несомненный интерес для разработчиков и исследователей систем защиты информации. Вместе с тем, трудность заключается в том, что современные шифры должны преобразовывать данные в последовательности, неотличимые от случайных настолько, насколько это возможно. Причем это одно из главных требований, предъявляемых к современным блоковым шифрам, – неотличимость зашифрованной последовательности от случайной, даже тогда, когда шифруемые данные заведомо не случайны. В частности, если на вход алгоритма шифрования подается текст на естественном языке, то на выходе он должен выглядеть как случайная последовательность. Заметим, что это условие предъявлялось Национальным институтом стандартов и технологии США (NIST) к блоковому шифру при организации конкурса на блоковый шифр 21-го века.

Как известно, для создания и хранения текстовой информации используются различные цифровые форматы (такие как “txt”, “tex”, “html”, “doc”, rtf”, “pdf”). В связи с этим, возникает вопрос о зависимости результата шифрования текста от формата, в котором он был представлен. Решение этой задачи представляет несомненный интерес для практической криптографии, разработчиков и исследователей систем защиты информации. Поэтому исследование влияния исходного формата шифруемых данных на возможность их различения в зашифрованном виде от случайных последовательностей важно с практической точки зрения.

Генерация “истинно” случайных чисел - одна из главных проблем возникающих при реализации любой криптосистемы. Огромное количество приложений уязвимо из-за предсказуемости генерируемых чисел. Дж. Клейнен в 1978 году приходит к выводу, что "все еще нет высококачественного генератора псевдослучайных чисел". Известные специалисты Льюис Кэрролл и Д. Марсалья указывают: "Многие широко распространенные генераторы фактически непригодны". Предостережение о распространенности плохих генераторов делает также И.М.Соболь. Вместе с тем, любое тестирование генераторов криптографически стойких случайных последовательностей остается частичным. Поэтому Ермаков С. М. и Михайлов Г. А. отмечают, что кроме специального статистического тестирования "... чрезвычайно важна проверка с помощью решения типовых задач, допускающих независимую оценку результатов аналитическими или численными методами. Можно сказать, что представление о надежности псевдослучайных чисел создается в процессе их использования с тщательной проверкой результатов всегда, когда это возможно".

Целью данной диссертационной работы является построение и разработка эффективных алгоритмов и комплекса программ для тестирования случайных и псевдослучайных чисел, а также их применение к следующим задачам:

1. Экспериментальное исследование генераторов криптографически стойких псевдослучайных последовательностей, созданных на основе современных алгоритмов шифрования, а также разработка комплекса программ для их тестирования.
2. Разработка и исследование эффективных численных методов и алгоритмов различения зашифрованных текстов на естественных языках от случайных последовательностей.
3. Экспериментальное исследование влияния формы представления данных на возможность их различения в зашифрованном виде.

Методы исследования. В работе были использованы вычислительные эксперименты на ЭВМ, моделирование, математическая статистика и теория построения эффективных алгоритмов и программ.

Научная новизна результатов работы. В диссертации получены следующие новые результаты:

1. Проведены экспериментальные исследования нового статистического критерия – “адаптивный критерий χ^2“, показана целесообразность его применения для статистической проверки случайных последовательностей и даны практические рекомендации по его применению.

2. Предложен эффективный алгоритм статистической проверки “качества” криптографически стойких псевдослучайных последовательностей с помощью разработанного комплекса программ.

3. Впервые предложен метод различения зашифрованных текстов на естественных языках от случайных последовательностей, позволяющий, с помощью разработанного комплекса программ, различать зашифрованные тексты при сравнительно небольших объемах данных (от 100 килобайт и выше). Отметим, что для популярного статистического критерия χ^2 требуется для решения этой задачи в тысячи раз больший объем данных.

4. Исследовано влияние формата цифровых текстов (txt, rtf, doc, pdf) и их предварительного сжатия на возможность различения в зашифрованном виде на русском, английском и итальянском языках.

Практическая ценность.

1. Разработаны алгоритмы и комплекс программ для тестирования случайных и псевдослучайных последовательностей, эффективность которых существенно выше, чем у ранее известных.

2. Разработан метод и предложен комплекс программ, позволяющий надежно различать зашифрованные тексты на естественных языках от случайных последовательностей, начиная с длины 100 килобайт.

3. Исследовано влияние формата цифровых текстов на естественных языках на возможность их различения в зашифрованном виде.

Связь с государственными программами. Работа выполнена в рамках проектов № 99-01-00586, № 03-01-00495 поддержанных РФФИ и INTAS-00-738 “Эффективное кодирование источников информации и связанные проблемы”.

Апробация работы. Основные результаты диссертации докладывались на международных научных мероприятиях: международная конференция по теории информации ISIT-2003 (Yokohama, Japan, 2003), Китайско-российский форум молодых ученых в Пекине – 2003 г. (Пекин, НаньЦзин, У си, Шан Хай, Китай, 2003), международная конференция Вычислительные технологии и математическое моделирование в науке, технике и образовании ВТММ-2002 (Алматы, Казахстан, 2002), Четвертое международное совещание по электронным публикациям El-Pub-99 (Новосибирск, 1999), а также на объединенных семинарах института вычислительных технологий СО РАН, кафедры математического моделирования НГУ, кафедры вычислительных технологий НГТУ.

Личный вклад. Выносимые на защиту результаты получены соискателем лично. В опубликованных совместных работах участие автора заключалось в разработке и исследовании алгоритмов, их программной реализации и численном моделировании. Постановка и аналитическое исследование задач осуществлялись совместными усилиями соавторов при непосредственном участии соискателя.

Публикации. По теме диссертации автором опубликовано 6 работ.

На защиту выносятся результаты экспериментальных исследований и совокупность вычислительных алгоритмов тестирования псевдослучайных и случайных чисел, в том числе и предназначенных для различения зашифрованных текстов.

Объем и структура диссертации. Диссертационная работа изложена на 119 страницах и состоит из введения, трех глав и заключения. Иллюстрационный материал включает 10 рисунков. Список литературы состоит из 43 наименований.

СОДЕРЖАНИЕ РАБОТЫ

Во введении обосновывается актуальность темы диссертационной работы, сформулированы цель работы и исследований, описываются полученные результаты, дано краткое изложение диссертации по главам.

В первой главе приведены известные результаты и сведения, используемые в дальнейшем. Кратко даются общие сведения по изучаемым в диссертации вопросам, приводится описание адаптивного критерия χ^2 используемого в работе.

Критерий хи-квадрат (χ^2) один из наиболее популярных тестов проверки гипотез, который широко применяется в экономике, биологии, криптографии и многих других областях. Например, критерий хи-квадрат используется для проверки генераторов случайных чисел и пригодности блоковых шифров для использования их как генераторов случайных чисел. В таких прикладных программах число категорий (и, следовательно, число степеней свободы χ^2 распределения) очень большое и, таким образом, объем выборки должен быть также большим. Значит, в этом случае, вычисления статистики хи-квадрат требуют много времени. Кроме того, часто практически трудно получить такие большие выборки и χ^2 не может применяться.

Предлагается новый метод, который назван *адаптивный критерий хи-квадрат*. Показано, что новый критерий применим, когда объем выборки намного меньше, чем это требуется для обычного критерия хи-квадрат. Остановимся кратко на основной идее метода. Пусть существует гипотеза H_0, которая утверждает, что символы из некоторого алфавита $A = \{a_1, a_2, ..., a_k\}$, $k > 2$, распределены равномерно (т.е. $p(a_1) = p(a_2) = ... = p(a_k) = 1/k$) против альтернативной гипотезы H_1, что истинное распределение неравномерно. Пусть дана выборка, которая может быть использована для проверки. Выборка делится на две части, которые называются обучающей и контрольной (проверочной). Обучающая выборка используется для оценки частоты встречаемости символов (букв). После этого буквы алфавита A объединяются в подмножества (классы) $A_1, A_2, ..., A_s$, $s \geq 2$, таким способом, что во-первых, в один класс группируются символы с близкими (или равными) частотами встречаемости и, во-вторых, s гораздо меньше чем k (скажем, $k = 2^{20}$, $s = 2$). Затем, классы $\{A_1, A_2, ..., A_s\}$ рассматриваются как новый алфавит и новая гипотеза $\widehat{H}_0 : p(A_1) = |A_1|/k, p(A_2) = |A_2|/k, ... p(A_s) = |A_s|/k$ и альтернативная гипотеза, которая является отрицанием $\widehat{H}_0$, проверяется на второй (контрольной) части выборки. Очевидно, если H_0 истинно, тогда $\widehat{H}_0$ также истинно и, если $\widehat{H}_1$ истинно, тогда H_1 истинно. Именно поэтому новый тест может быть использован для проверки первоначальных H_0 и H_1. Идея такой схемы довольно проста. Если H_1 истина, тогда есть символы с относительно большими и относительно малыми вероятностями. Вообще говоря, высоковероятные символы будут иметь относительно большие частоты встречаемости и будут накоплены в некоторых классах A_i, тогда как низко вероятные символы будут накоплены в других классах. Именно поэтому, это отличие может быть найдено по проверочной выборке. Важно отметить, что уменьшение числа категорий от k до меньшего s, может существенно увеличить мощность критерия и, поэтому, может

существенно уменьшить задаваемый объем выборки. Более точно, показано, что объем выборки, может быть уменьшен в $\sqrt{k}$ раз, что может быть важно, когда k большое.

Во второй главе диссертации, состоящей из 2 параграфов, рассматриваются экспериментальные исследования эффективных генераторов криптографически стойких псевдослучайных последовательностей, созданных на базе современных алгоритмов шифрования.

В 1994 году впервые появилось описание алгоритма шифрования RC5, а в 1998 г. – алгоритма RC6. Параграф 2.1 посвящен исследованию криптографических алгоритмов RC5 и RC6 как датчиков случайных чисел. По сравнению с традиционными алгоритмами датчиков псевдослучайных чисел эти криптографические алгоритмы обладают многими достоинствами, одним из которых является очень высокое быстродействие. Авторы алгоритмов рекомендовали их для генерации последовательностей случайных чисел. Однако на время проведения экспериментальных исследований каких-либо статистических данных по исследованию или применению генераторов псевдослучайных чисел основанных на этих алгоритмах мы не нашли. Таким образом, экспериментальные исследования решали задачу статистического анализа генераторов псевдослучайных чисел, базирующихся на криптографических алгоритмах шифрования RC5, RC6 и разработки рекомендаций по их применению. Для исследований использовался критерий Пирсона χ^2, являющийся одним из самых распространенных и эффективных тестов.

На основании проведенных экспериментальных исследований был сделан вывод о том, что алгоритмы шифрования RC5 и RC6 можно использовать в качестве генераторов псевдослучайных чисел. Однако, полученные результаты тестирования показали, что для RC5 требуется специальный алгоритм входных данных, при реализации которого значительно улучшаются его статистические свойства. Временные характеристики алгоритма RC5 уступают характеристикам алгоритма RC6. С помощью RC6 можно получить последовательность случайных чисел значительно большей длины, чем с помощью алгоритма RC5 (период последовательности 2^{128} против 2^{32}). Выходная последовательность при одном обращении к алгоритму для RC6 (128 бит) в два раза больше, чем для RC5. Учитывая вышесказанное, мы рекомендовали для получения последовательности случайных чисел в качестве генератора использовать криптографический алгоритм шифрования RC6 (с количеством раундов $r = 6$ или $r = 11$).

Для реализации алгоритма шифрования RC6 в качестве генератора псевдослучайных чисел на языке Java предлагается RC6Key.class на http://cins.ict.nsc.ru/Class/RC6Key.class. Для возможности реализации рассматриваемых криптографических алгоритмов шифрования на других языках программирования, в конце параграфа дано их полное описание, включая алгоритм расширения ключа.

В параграфе 2.2 рассматриваются исследования криптографических алгоритмов, победителя и финалистов конкурса на блоковый “шифр 21-го века”, проведенного NIST (США) в 1999-2001 г.г., AES, RC6 и MARS с помощью нового статистического метода - адаптивный критерий χ^2.

Экспериментальные исследования эффективности генераторов псевдослучайных чисел, проведенные с алгоритмами шифрования RC5 и RC6 (см. параграф 2.1), показали неплохие результаты. Однако эти исследования были ограничены длиной слова (или блока) в 26 бит, что связано с требованиями, для критерия χ^2, на длину проверяемой последовательности и доступными вычислительными ресурсами. Новый статистический критерий - адаптивный критерий χ^2 позволяет, довольно существенно, снизить ограничения на выбранную длину слова и улучшить качество проверки случайной

последовательности. Расчет минимально необходимой длины проверяемой последовательности для адаптивного критерия χ^2 проводится по формуле (1).

$$\frac{N \times k}{2^b} \times (N \times (1-k)) \geq 5, \quad (1)$$

где N – длина проверяемой последовательности в байтах, b – длина выбираемого слова в битах, k – коэффициент отношения обучающей выборки от N. Например, для традиционного критерия χ^2 при тридцати двух битной выборки, необходимая минимальная длина проверяемой последовательности 85899345920 байт, а для адаптивного критерия χ^2 – 1172308 байт (где $k = 1/2$).

В данном параграфе представлены результаты экспериментальных исследований датчиков псевдослучайных чисел созданных с помощью криптографических алгоритмов AES, RC6 и MARS. Испытания полученных псевдослучайных последовательностей проводились на длинах слов (букв) b = {8, 16, 24, 32, 40, 48} бит. На первом, подготовительном этапе, с “CDROM-а случайных чисел” (“The Marsaglia Random Number CDROM”), созданного профессором университета Флориды Д. Марсалья (George Marsaglia), были произвольно выбраны сто 128 битных “случайных” ключей. Подготовленный комплекс программ реализации алгоритмов шифрования AES, Mars и RC6 использовался для получения последовательностей случайных чисел необходимых в исследовании. Для каждого создаваемого файла использовался один из последовательно выбираемых подготовленных 128 битных “случайных” ключей и на вход, текущего в данный момент алгоритма шифрования, подавались числа y_i = {0, 1, …, n}. На выходе алгоритма мы получали файл размерностью 460 мегабайт. Таким образом, для исследования было подготовлено 300 файлов или по 100 зашифрованных последовательностей для каждого выбранного алгоритма шифрования.

Рассмотрим разработанный алгоритм, который можно представить двумя основными частями: обучения и проверки (контрольной).

Обучение. По заданной для испытания длине файла N байт (b, k и N являются параметрами программы) определяется, удовлетворяет ли она условию (1). Через заданное N определяем количество слов (здесь, под словом будем понимать битовую последовательность Integer, если $b < 32$ и Double, если $b \geq 32$) входящих заданную последовательность $L = N/(w/8)$, где w = 64 для выборки $b \geq 32$ и w = 32 для выборки $b < 32$. Затем, через вычисленное количество слов L находим длину обучающей L_e и контрольной L_c частей в словах по формуле.

$$L_e = L \times k$$
$$L_c = L \times (1-k)$$

Создаются две одномерные таблицы T и Ys размерностью I вычисляемой с помощью найденных значений L_e и L_c по формуле (2). Величина I будет равна максимально возможному числу букв (здесь и далее в этом параграфе, под буквой будем понимать битовую последовательность равную b битам из алфавита 2^b букв), для заданной размерности b бит, из наибольшей заданной обучающей или контрольной последовательности.

$$\begin{cases} I = L_e w / b, & если \quad L_e \geq L_c \\ I = L_c w / b, & если \quad L_c > L_e \end{cases} \quad (2)$$

В ячейки таблицы T из обучающей части $m_i = \{x_1, x_2, ..., x_{L_e}\}$, заданного для проверки файла, последовательно выбираются буквы заданной размерностью выборки b бит. После

прохождения m_i, для оптимизации дальнейшей обработки данных из T в алгоритме используется сортировка Шелла.

По результатам обработки обучающей части подготавливается таблица *Ys*, которая будет использования в контрольной части. Для этого из таблицы T последовательно выбирается по одной букве и, если она ранее не встречалось, ее значение заноситься в первую же свободную ячейку *Ys*. Таким образом, в таблице *Ys* будут накоплены буквы (буквы в таблице будут располагаться по возрастанию), попавшие в обучающую область m_i.

Проверка. Для контрольной части алгоритма используется вторая половина исходной последовательности $n_i = \{x_{L_e+1}, x_{L_e+2}, ...x_{L_c}\}$, таблица T и, подготовленная на этапе обучения, таблица *Ys*. В ячейки таблицы T из контрольной части $n_i = \{x_{L_e+1}, x_{L_e+2}, ...x_{L_c}\}$, заданного для проверки файла, последовательно выбираются буквы заданной размерностью выборки b бит. Для оптимизации дальнейшей обработки данных в алгоритме используется сортировка Шелла букв в таблице T .

Дополнительно создаются две таблицы V и P с размерностью 2 - равной количеству создаваемых новых классов. Из подготовленной таблицы T последовательно выбираются буквы, полученные из контрольной части исходной последовательности, и сравниваются с буквами, попавшими в обучающую часть. Если в таблице *Ys* будет найдена буква равная выбранной, увеличивается количество попаданий из контрольной выборки во второй класс $V[2] = V[2] + 1$, в противном случае, увеличивается счетчик $V[1] = V[1] + 1$ для первого класса.

Подсчитывается общее количество строк (ячеек) с буквами в таблице *Ys* и полученный результат *newIndex* заносится в таблицу $P[2] = newIndex$. Первая ячейка таблицы $P[1]$ будет содержать разницу между максимально возможным появлением числа букв для выборки b и реально встретившимися в обучающей части (m_i) $P[1] = 2^b - P[2]$.

В завершающей части алгоритма, по результатам выборки из n_i оцениваются частоты попаданий букв (таблица V) для контрольной части и подсчитывается величина χ^2 для двух классов.

$$\chi^2 = \sum_{i=1}^{c} \frac{(V_i - nP_i)^2}{nP_i}, \qquad (3)$$

где V_i количество элементов, попавших в i класс; n – общее количество букв в выборке n_i (V_1 V_2); P_i – вероятность попадания в i класс результата испытаний по контрольной части (P_i – количество букв, относящихся к "i" классу, деленное на 2^b), c – количество выбранных классов.

При большой исходной выборке (N) и малой длине b разность $P[1] = 2^b - P[2]$ может равняться нулю. Это означает, что в обучающей области встретились все возможные значения букв из области {0, 1, …, $2^b - 1$}. В этом случае используется алгоритм с тремя классами (причем алгоритм с тремя классами может быть использован и для первого варианта, когда b большое, а $2^b - P[2] > 0$). Две таблицы V и P создаются с размерностью равной 3 - количеству создаваемых новых классов и дополнительная таблица *Yb* размерностью *newIndex*. После обучающей части, алгоритм которой не меняется, выполняется дополнительная процедура создания классов – "Dispersion".

Находим максимальное *Max* и минимальное *Min* значение выбранных по обучающей области букв из *Ys*, после чего, вычисляется дисперсия D необходимая для создания новых классов по формуле, выбранной после многочисленных экспериментов, $D = Min + (Max - Min)/3$. (На самом деле алгоритмов определения дисперсии может быть

очень много и оптимальные, для каждого распределения, можно найти экспериментальным путем).

Из созданной на этапе обучения таблицы *Ys* последовательно выбираются буквы и помечаются, к какому классу они относятся. Если значение в ячейки таблицы *Ys* меньше или равно найденному значению *D*, в идентичную ячейку таблицы *Yb* заносится единица, признак отношения к первому классу, а счетчик отношения для данного класса в таблице *P* увеличивается на единицу *P*[1] = *P*[1] + 1. Все выбранные значения из *Ys*, удовлетворяющие неравенству $D < Ys[i] \le D + (Max - Min)/3$, будут отнесены ко второму классу (в идентичные ячейки таблицы *Yb* заносится 2 и счетчик для данного класса в таблице *P* увеличивается на единицу *P*[2] = *P*[2] + 1 для каждого события). Значения из *Ys*, для которых неравенство $Ys[i] > D + (Max - Min)/3$ является истиной, помечаются как имеющие отношение к третьему классу (в идентичные ячейки *Yb* заносится 3 и *P*[3] = *P*[3] + 1 для каждого случая).

После обработки всей таблицы значений *T* по полученным результатам (выборки из n_i) оцениваются частоты попаданий букв (таблица *V*) для контрольной части и подсчитывается величина χ^2 (3) для трех классов. Где V_i количество элементов, попавших в *i* класс; *n* – общее количество букв в выборке n_i (V_1 + V_2 + V_3); P_i – вероятность попадания в *i* класс результата испытаний по контрольной части (P_i – количество букв, относящихся к "*i*" классу, деленное на 2^b).

Наконец, результаты проведенных экспериментальных исследований использования современных алгоритмов шифрования в качестве генераторов криптографически стойких псевдослучайных последовательностей, позволяют сделать вывод, что современные криптографические алгоритмы шифрования AES, Mars и RC6 можно использовать как генераторы псевдослучайных чисел.

На основании проведенных экспериментов, заключаем, что мощность нового метода адаптивный критерий χ^2 существенно выше, чем у традиционного критерия Пирсона χ^2. Новый метод более эффективен во многих приложениях, в частности связанных с криптографией, где *b* велико, а объем доступной выборки ограничен.

Третья глава диссертации посвящена задаче построения статистического метода для алгоритма различения зашифрованных текстов на естественных языках и случайных последовательностей.

В параграфе 3.1 предлагается и исследуется один из первых результативных вариантов алгоритма, позволяющий достаточно надежно различать зашифрованные тексты на русском, английском и итальянском языках, в формате данных "*txt*", от случайных последовательностей для длин выборки *N* = {512000, 1024000, 2048000} байт. Конструкция предлагаемого алгоритма базируется на двух этапном подходе. Сначала исследуемая последовательность разбивается на (под)слова по 24 бита, что соответствует алфавиту $A = \{a_1, \ldots, a_k\}$, где $k = 2^{24}$, которая затем делится на две равные части $m_i = \{x_1, x_2, ..., x_{N_i/2}\}$ и $n_i = \{x_{N_i/2+1}, x_{N_i/2+2}, ...x_{N_i}\}$ – обучающую и контрольную. По результатам прохождения всей обучающей выборки $m_i = \{x_1, x_2, ..., x_{N_i/2}\}$ алфавит из 2^{24} букв разбивается на три класса ((под)алфавита), содержащих буквы с частотами встречаемости, оцененными по обучающей части. Для этого в таблицу *T* размерностью 2^{24} (количество ячеек таблицы равно количеству слов алфавита *A*) накапливается счетчик попаданий слова (буквы) из алфавита *A*, $T[a_i(m_i)] = T[a_i(m_i)] + 1$, другими словами, для каждой встреченной буквы в соответствующей ячейке таблицы счетчик увеличивается на единицу. Затем, по данным, накопленным в таблице *T*, определяется максимальное число попаданий $M_{max} = \max(T[a_i(m_i)])$, т.е. счетчик для слова, которое встретилось наибольшее количество раз в обучающей части, с помощью которого вводится новый коэффициент среднего числа попаданий для трех классов $K_{mean} = M_{max}/3$. Используя этот коэффициент,

проводится группировка (или объединения по классам) букв, с близкой частотой встречаемости, в новые “супербуквы” {A_1, A_2, A_3}. На втором этапе используется таблица T подготовленная на шаге обучения и две новые таблицы V и P с размерностью равной количеству созданных новых классов. Подсчитывается общее количество строк (ячеек) в таблице T для каждого класса cl = {1, 2, 3} и результат заносится в таблицу $P[cl]$. Из $n_i = \{x_{N_i/2+1}, x_{N_i/2+2}, ...x_{N_i}\}$, аналогично этапу обучения, последовательно выбираются слова по 24 бита $(x_{N_i/2+1}, ..., x_{N_i})$ и по содержимому ячейки таблицы $T[a_i(n_i)]$ определяется, к какому классу относится выбранное слово. В таблице V суммируется количество попаданий в данный класс $V[T[a_i(n_i)]] = V[T[a_i(n_i)]] + 1$, после чего оцениваются частоты попаданий букв ($V_{cl}(n_i)$) из алфавита A для контрольной части, подсчитывается статистика χ^2 для 3 классов и проверяется гипотеза о случайности зашифрованного текстового файла.

В параграфе 3.2 предлагается новый алгоритм, позволяющий достаточно надежно различать зашифрованные тексты на русском, английском и итальянском языках, для четырех наиболее известных форматов данных: “*txt*”, “*rtf*”, “*doc*”, “*pdf*”, от случайных последовательностей начиная с длины 100 килобайт. Для этого зашифрованный файл, некоторой длины N байт (исходная выборка), разбивается на слова a по 32 бита, что соответствует алфавиту A = {a_1, …, a_k}, где $k = 2^{32}$. Полученная последовательность исследуется "на случайность" по четырем значениям N = {512000, 307200, 204800, 102400} байт. Исходная выборка разбивается, как и в предыдущем варианте, на две части $m_i = \{x_1, x_2, ..., x_{N_i/2}\}$ и $n_i = \{x_{N_i/2+1}, x_{N_i/2+2}, ...x_{N_i}\}$ – обучающую и контрольную. Затем подсчитывается частота встречаемости букв из A в обучающей выборке. По результатам обработки обучающей выборки алфавит из 2^{32} букв разбивается на два класса (A_0, A_1), содержащих буквы с частотами встречаемости, соответственно, 0 и 1 оцененными по обучающей части. Затем, по контрольной выборке для полученных двух классов, рассматриваемых как отдельные "супербуквы" A_0, и A_1 проверялась гипотеза

$$H_0 = \left\{ P\{x \in A_0\} = \left|A_0\right| / 2^{32}, P\{x \in A_1\} = \left|A_1\right| / 2^{32} \right\}$$

против гипотезы H_1, являющейся отрицанием H_0. (Здесь x – элемент выборки, или в нашем случае, слово длины 32 двоичных знака из проверочной выборки).

Это задача проверки гипотезы о параметре биноминального распределения. Она хорошо известна в математической статистике [2], и для ее проверки можно применять не только критерий χ^2, но и другие методы, применимые при малых значениях $P\{x \in A_0\}$ (см. [2]).

В данной работе мы использовали критерий, базирующийся на непосредственном использовании биноминального распределения [2]. Для его описания мы определим величину

$$\pi = \left|A_1\right| / 2^{32}$$

и обозначим через l количество 32 – битовых слов, а через x количество элементов из контрольной выборки, попавших в множество A_1. Пусть α - заданный уровень значимости критерия. Определим P_1 как наибольшее значение P, удовлетворяющее неравенству

$$\sum_{k=x}^{l} \binom{l}{k} P^k (1-P)^{l-k} \le \alpha / 2,$$

а P_2 как наименьшее значение P, удовлетворяющее неравенству

$$\sum_{k=0}^{x} \binom{l}{k} P^k (1-P)^{l-k} \le \alpha / 2 .$$

Если $\pi \in (P_1, P_2)$, то гипотеза H_0 принимается, в противном случае H_0 отвергается.

В **заключении** кратко формулируются основные результаты диссертационной работы и сказано несколько слов о перспективах исследования.

Основные результаты и выводы

1. Разработан метод различения зашифрованных текстов на естественных языках от случайных последовательностей, при небольших объемах данных. Впервые предложен метод позволяющий решить эту задачу при длине исследуемой последовательности от 100 килобайт, тогда как ранее известные методы применимы только при объеме данных в сотни раз большем. Кроме того, исследовано влияние формата цифровых текстов на возможность их различения в зашифрованном виде.

2. Показана целесообразность использования статистического критерия – "адаптивный критерий χ^2" для статистической проверки случайных и псевдослучайных последовательностей и его эффективность в приложениях, в частности связанных с криптографией, где объем доступной выборки может быть ограничен. На его основе разработаны алгоритмы и программы тестирования случайных и псевдослучайных чисел.

3. Исследованы датчики псевдослучайных чисел созданные на базе современных криптографических алгоритмов шифрования AES, Mars, RC6 и RC5. Показано, что алгоритм шифрования RC5 (без подбора входных данных) нельзя использовать для получения псевдослучайных чисел. Остальные три алгоритма шифрования можно использовать как генераторы псевдослучайных чисел для получения "качественной" криптографически стойкой случайной последовательности.

Содержание диссертации отражено в следующих работах:

1. Рябко Б.Я., Стогниенко В.С., Шокин Ю.И. Экспериментальные исследования эффективности генераторов псевдослучайных чисел, базирующихся на криптографических алгоритмах RC5 и RC6 // Вычислительные технологии. – 2000. - Т. 5, № 6. – С. 70-79.
2. Рябко Б.Я., Стогниенко В.С., Шокин Ю.И. Экспериментальные исследования статистических свойств зашифрованных текстов на естественных языках // Вычислительные технологии. – 2003. - Т. 8. – С. 100-108.
3. Рябко Б.Я., Стогниенко В.С., Шокин Ю.И. Адаптивный критерий χ^2 для различения близких гипотез при большом числе классов и его применение к некоторым задачам криптографии // Проблемы передачи информации. – М.: РАН, 2003. - Т. 39, - Вып. 2. – С. 53-62.
4. Стогниенко В.С. Экспериментальные исследования статистических свойств современных блоковых шифров и их применение к одной из задач криптографии// По материалам Международной конференции ВТММ-2002. - Новосибирск-Алматы. - 2002 – Т. 5. - С. 182-186
5. Ryabko, Stognienko, Shokin. Adaptive chi-square test and its application to some cryptographic problems, Journal of statistical planning and inference (JSPI). – 2004. - v. 123, n. 2. - pp. 365-376.
6. Ryabko B.Ya., Stognienko V.S., Shokin Yu. I.. A new testing for random numbers and its application to some cryptographic problems // International Symposium on Information Theory (2003 IEEE) (Yokohama, Japan, June 29 – July 4, 2003). – Yokohama, 2003. – P. 338.