МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

Кафедра математического моделирования и анализа данных

Долгая Александра Валерьевна

Анализ и оценивание параметров генераторов на основе регистров сдвига

Дипломная работа

Научный руководитель:

В.Ю. Палуха

доцент кафедры ММАД,

кандидат физико-математических наук

Допущена к защите
«___»_____20__г.

Заведующий кафедрой математического

моделирования и анализа данных

кандидат физико-математических наук,

профессор, В.И. Малюгин

Минск, 2025

ОГЛАВЛЕНИЕ

РЕФЕРАТ

Дипломная работа включает 49 страниц, 10 рисунков, 2 таблицы, 12 источников, 5 приложений.

Ключевые слова: ПРОРЕЖИВАЮЩИЙ ГЕНЕРАТОР, САМОСЖИМАЮЩИЙ ГЕНЕРАТОР, РЕГИСТР СДВИГА, ПЕРИОД ПОСЛЕДОВАТЕЛЬНОСТИ, ЗАВИСИМОСТЬ РАНГА ОТ ПОРЯДКА МАТРИЦЫ, GF(2).

Объект исследования: выходные последовательности генераторов.

Цель работы: нахождение зависимостей рангов матриц от их порядка путем формирования матриц над полем GF(2) из выходных последовательностей прореживающего и самосжимающего генераторов.

Методы исследования: математическое моделирование, линейная алгебра над полем GF(2), статистический анализ, алгоритмическая реализация генераторов и расчетов, графическая интерпретация результатов.

Полученные результаты и их новизна: разработаны и протестированы алгоритмы генерации выходных последовательностей для прореживающего и самосжимающего генераторов на основе LFSR, вычислены зависимости рангов матриц от их порядка, построенных из выходных битов генераторов, что дает представление об их структуре и нелинейных свойствах, новизна заключается в комбинированном применении криптографического анализа и линейной алгебры на GF(2) для оценки качества генераторов, что является полезным при проектировании стойких криптографических систем.

Достоверность материалов и результатов дипломной работы: все полученные данные основаны на строго формализованных математических моделях, примеры реализации последовательностей подтверждают работу генераторов согласно теории, результаты прошли верификацию через статистические тесты, применение различных примитивных порождающих полиномов усиливает обоснованность вывод.

Область применения: информационная безопасность, инженерия, научно-исследовательские институты.

РЭФЕРАТ

Дыпломная праца ўключае 49 старонак, 10 малюнкаў, 2 табліцы, 12 крыніц, 5 прыкладанняў.

Ключавыя словы: ПРАРЭДЖВАЛЬНЫ ГЕНЕРАТАР, САМАСЦІСКАЛЬНЫ ГЕНЕРАТАР, РЭГІСТР ЗРУХУ, ПЕРЫЯД ПАСЛЯДОЎНАСЦІ, ЗАЛЕЖНАСЦЬ РАНГУ АД ПАРАДКУ МАТРЫЦЫ, GF(2).

Аб'ект даследавання: выхадныя паслядоўнасці генератараў.

Мэта працы: знаходжанне залежнасцяў рангаў матрыц ад іх парадку шляхам фарміравання матрыц над полем GF(2) з выходных паслядоўнасцяў праяреживающего і самосжимающего генератараў.

Метады даследавання: матэматычнае мадэляванне, Лінейная алгебра над полем GF (2), статыстычны аналіз, алгарытмічная рэалізацыя генератараў і разлікаў, графічная інтэрпрэтацыя вынікаў.

Атрыманыя вынікі і іх навізна: распрацаваны і пратэставаны алгарытмы генерацыі выходных паслядоўнасцяў для праяреживающего і самосжимающего генератараў на аснове LFSR, вылічаныя залежнасці рангаў матрыц ад іх парадку, пабудаваных з выходных бітаў генератараў, што дае ўяўленне аб іх структуры і нелінейных уласцівасцях, навізна заключаецца ў камбінаваным ўжыванні крыптаграфічнага аналізу і лінейнай алгебры на GF(2) для ацэнкі якасці генератараў, што з'яўляецца карысным пры праектаванні стойкіх крыптаграфічных сістэм.

Дакладнасць матэрыялаў і вынікаў дыпломнай працы: усе атрыманыя дадзеныя заснаваныя на строга фармалізаваных матэматычных мадэлях, прыклады рэалізацыі паслядоўнасцяў пацвярджаюць працу генератараў паводле тэорыі, вынікі прайшлі верыфікацыю праз статыстычныя тэсты, прымяненне розных прымітыўных спараджаюць полиномов ўзмацняе абгрунтаванасць выснову.

Вобласць ужывання: інфармацыйная бяспека, інжынерыя, навукова-даследчыя інстытуты.

ABSTRACT

Graduate work: 49 pages, 10 pictures, 2 tables, 12 sources, 5 applications.

Key words: PROGRAMMING GENERATOR, SELF-CONNECTING GENERATOR, ADVANCE REGISTER, LETTER PERIOD, RANGE RANGE LOCALITY OF MATRIX, GF(2).

Object of study: output sequences of oscillators.

Objective: finding the dependence of matrix ranks on their order by forming matrices over the field GF(2) from output sequences of thinning and self-compressing generators.

Methods of research: mathematical modelling, linear algebra over the field GF(2), statistical analysis, algorithmic implementation of generators and calculations, graphical interpretation of the results.

Results: the first and second moments were found using the recurrent formula and the derivative of the generating function, the results were compared based on graphs obtained by modeling, and the probabilities of occurrence and non-occurrence of a series of events were found.

Reliability of materials: all obtained data are based on strictly formalised mathematical models, examples of sequences implementation confirm the work of generators according to the theory, the results have been verified through statistical tests, application of various primitive generating polynomials strengthens the validity of the conclusion.

The field of application: information security, engineering, research institutes.

ВВЕДЕНИЕ

1.1 Вступление

Анализ генераторов на основе регистров сдвига по их выходным последовательностям – одна из важных задач в сфере криптографии и информационной безопасности. Генераторы на основе регистров сдвига (ГРС) широко используются в различных приложениях, включая шифрование данных, формирование ключей, генерацию случайных чисел и обеспечение конфиденциальности в сетях.

ГРС представляют собой последовательные логические элементы, связанные в цепочку, с каждым элементом, имеющим два состояния – 0 и 1. Они обеспечивают генерацию последовательностей битов на основе циклического сдвига значений внутренних регистров. Выходные последовательности ГРС считаются псевдослучайными, то есть имеют статистические свойства, которые делают их неотличимыми от истинно случайных.

Однако, не все ГРС достаточно безопасны и надежны. Некоторые из них могут иметь слабости в виде периодичных или непредсказуемых шаблонов в выходных последовательностях, что делает их уязвимыми для атак. Поэтому анализ и оценка качества выходных последовательностей ГРС является критической задачей для обеспечения криптографической стойкости и безопасности систем, использующих такие генераторы.

Целью данной дипломной работы является рассмотрение различных методов и подходов к анализу генераторов на основе регистров сдвига по их выходным последовательностям, а также выявления закономерностей между рангами и порядками матриц, сформированных из полученных последовательностей. Изучатся статистические тесты и метрики, которые помогают оценить равномерность, сложность и случайность выходных последовательностей. В данной работе будут рассматриваться два типа генераторов: прореживающий и самосжимающий, их принцип работы, сильные и слабые стороны, методы усовершенствования ГРС для повышения их стойкости.

Понимание анализа генераторов на основе регистров сдвига по их выходным последовательностям является важным инструментом для проектирования и выбора безопасных систем шифрования и защиты информации. Это также позволяет исследователям и специалистам по информационной безопасности вовлечься в обнаружение и исправление

потенциальных уязвимостей в существующих ГРС, обеспечивая более надежную защиту данных.

1.2 Определение основных понятий

1. Регистр сдвига (Shift Register):

Регистр сдвига – это устройство, собранное из триггеров, которые запоминает и перемещают данные. Триггеры связаны таким образом, что хранящиеся в них данные сдвигаются в выбранном направлении от одного триггера к другому, соседнему триггеру, по каждому тактовому импульсу. Регистры сдвига используются для преобразования данных из последовательного формата в параллельный, для обмена данными в последовательном коде, а также в качестве элементов задержки.

2. Псевдослучайные последовательности (Pseudorandom Sequences):

Псевдослучайные последовательности создаются генераторами для имитации статистических свойств случайных чисел. В контексте данной темы, генераторы на основе регистров сдвига используются для создания таких псевдослучайных последовательностей.

3. Линейные обратные связи (Linear Feedback Shift Register - LFSR):

LFSR – это регистр сдвига с линейной обратной связью, представляющий собой конечный автомат, который генерирует последовательность битов по рекуррентному правилу. Применяется для генерации псевдослучайных последовательностей битов.

4. Период генератора:

Период генератора - это количество шагов (или выходных битов), после которого генератор возвращается к начальному состоянию и начинает повторяться. Для эффективных генераторов важно иметь долгий период.

5. Криптостойкость (Cryptographic Strength):

Это свойство генератора, определяющее его устойчивость к криптоанализу, т.е. способность сопротивляться попыткам предсказания или восстановления его выходных последовательностей.

6. Спектральные характеристики:

Спектральные характеристики описывают распределение энергии или амплитуды в различных частотных компонентах выходных последовательностей генератора. Анализ этих характеристик помогает оценить качество генератора.

7. Корреляционные характеристики:

Оценка корреляционных свойств выходных последовательностей генератора позволяет измерить степень зависимости между выходными

битами ГРС и выявить наличие каких-либо статистических связей или шаблонов в последовательностях. Распространенными корреляционными характеристиками являются автокорреляция и попарная корреляция. Для оценки корреляционных характеристик ГРС существуют различные методы и статистические тесты.

8. Насыщение ранга (плато):

Насыщение ранга — это явление, при котором ранг матрицы перестает увеличиваться с ростом её размерности и достигает максимального значения, определяемого свойствами матрицы или ограничениями системы, которая её порождает.

9. Криптоанализ:

Криптоанализ - это процесс анализа криптографических систем с целью выявления их уязвимостей. Для генераторов на основе регистров сдвига, криптоанализ включает в себя поиск методов восстановления ключей или предсказания выходных последовательностей.

10. Прореживающий генератор:

Прореживающий генератор – это криптографический алгоритм, генерирующий псевдослучайную последовательность путем избирательного включения битов из одной последовательности (элементарной) на основе управляющей последовательности.

11. Самосжимающий генератор:

Самосжимающий генератор – это криптографический алгоритм, генерирующий псевдослучайную последовательность путем избирательного включения битов из одной базовой последовательности, создаваемой линейным регистром сдвига (LFSR).

12. Тест Колмогорова-Смирнова:

Критерий Колмогорова-Смирнова является непараметрическим критерием равенства непрерывных одномерных распределений вероятностей, который может быть использован для проверки того, была ли выборка получена из заданного эталонного распределения вероятностей.

13.Тест хи-квадрат:

Критерий хи-квадрат – любая статистическая проверка гипотезы, в которой выборочное распределение критерия имеет распределение хи-квадрат при условии верности нулевой гипотезы.

2. ТЕОРЕТИЧЕСКИЙ ОБЗОР

2.1 Криптографические генераторы

Современная криптография невозможна без случайных и псевдослучайных последовательностей x1, x2 , ...∈V ={0,1} [1]. Случайные последовательности генерируются при помощи физических генераторов, а псевдослучайные – при помощи программных генераторов. Практическую значимость имеют генераторы последовательностей, близких по своим свойствам к равномерно распределенной случайной последовательности (РРСП). РРСП – это случайная последовательность x1, x2, …, x_t, x_{t+1}, …, определенная на вероятностном пространстве (Ω, F, P) и удовлетворяющая двум требованиям:

C1: Для любого n∈ и произвольных значений индексов $1 \le t_1 < ... < t_n$ случайные величины x_{t_1} ,..., x_{t_n} независимы в совокупности.

C2: Для любого номера t∈N случайная величина x_t является бернуллиевой и имеет равномерное распределение вероятностей P{x_t =i}=1/2, i∈V ={0,1} .

Гипотезу о том, что выходная последовательность генератора { x_t } является равномерно распределенной, будем обозначать H* = {{ x_t } есть РРСП}, а альтернативу - H = H*.

Математической сущностью этой задачи является задача статистического распознавания генераторов случайных и псевдослучайных последовательностей, т.е. задача отнесения (классификации) наблюдаемой выходной последовательности генератора x1, x2, …, x_T ∈V ={0,1} некоторой конечной длительности T к одному из L ($2 \le L < +\infty$) классов Ω1, …, ΩL.

Генераторы могут быть разделены на следующие классы { Ω_i }: физические и программные генераторы; программные генераторы различных типов; программные генераторы одного типа, но с различными параметрами; программные генераторы с одинаковыми параметрами, но с различной инициализирующей информацией.

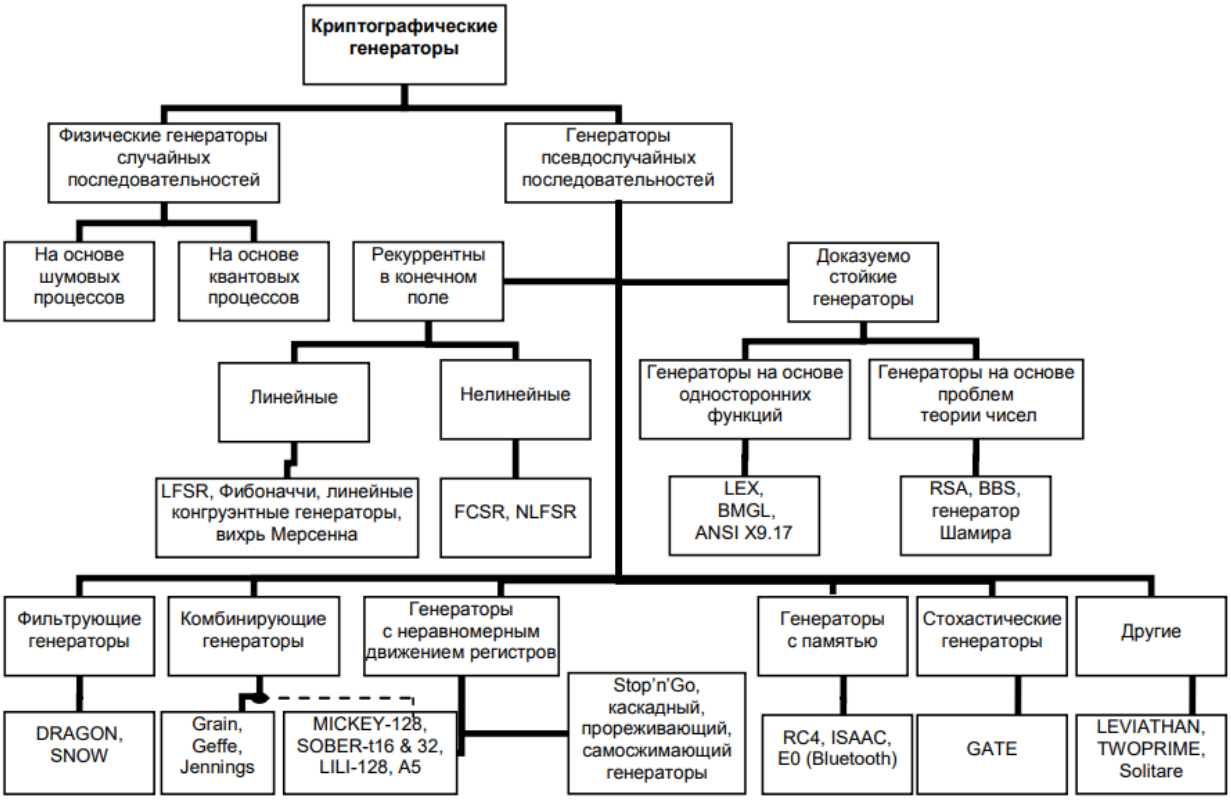
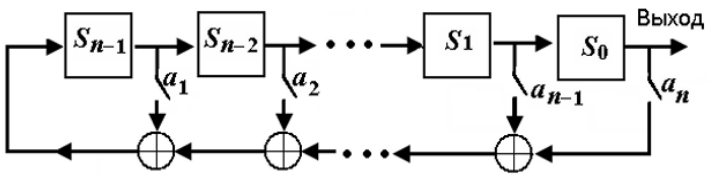


Рис. 1. Схема классификации криптографических генераторов

2.2 Регистры сдвига с линейной обратной связью

В технике один из наиболее распространенных методов генерации битовых последовательностей реализуется с помощью регистров сдвига с обратной линейной связью (англ. Linear Feedback Shift Register – LFSR).

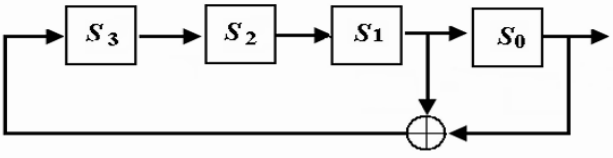


LFSR – микросхема с ячейками памяти, в которые записан один бит информации. Ячейки имеют по одному входу и выходу. Количество ячеек равно длине регистра. Вторая составляющая регистра – функция обратной связи. Для линейного регистра – это операция XOR над некоторыми его битами, которые называют отводами (точками съемки). Принцип работы регистра длины n с обратной линейной связью графически подан на рисунке. Регистр работает в дискретные моменты времени, в каждый из которых выполняются такие операции [9]:

- содержание самой правой ячейки S_0 «выталкивается» из регистра и формирует очередной элемент генерированной последовательности;

- содержание ячейки S_i перемещается в ячейку S_{i-1} , i= 0,1,..., n-1;
- новое содержание самой левой ячейки – это бит обратной связи, который равен сумме по модулю 2 битов ячеек S_0 , S_1 ,…, S_{n-1} , умноженных на коэффициенты a_1, a_2 ,..., a_n .

Если некоторые из коэффициентов a_1 , a_2 , ,..., a_n равны нулю, то соответствующие сумматоры $\oplus$ из цепи обратной связи исключают. В моменты времени t=0,1,2,... на выходе регистра генерируется последовательность x_0, x_1,.. Пример LFSR длины 4 с коэффициентами a_1= a_2=0, a_3= a_4=1 и начальным состоянием (S_0, S_1, S_2, S_3)=(1,0,1,1):



	$S_0S_1S_2S_3$	
$t=0$	1 0 1 1;	Генерируется последовательность
$t=1$	0 1 1 1;	1011110001001101
$t=2$	1 1 1 1;	
$t=3$	1 1 1 0;	
$t=4$	1 1 0 0;	
$t=5$	1 0 0 0	

Содержание ячеек называется заполнением (состоянием) регистра (в начальный момент – это начальное заполнение регистра или вектор начального состояния). При условии, что содержание всех ячеек равно нулю, состояние регистра называют нулевым. Если x_i (t), x(t) – заполнение i-ой ячейки памяти и выход регистра в момент времени t соответственно, то работу регистра описывают уравнения:

$$x_{i-1}(t+1) = x_i(t), \quad i = 0,1,...,n-1;$$
$$x_{n-1}(t+1) = f(x_0(t), x_1(t),..., x_{n-1}(t));$$
$$x(t+1) = x_0(t).$$

Выходная последовательность LFSR называется линейной рекуррентной последовательностью n-го порядка над полем GF(2) . Это последовательность битов, которые при всех $0 \leq i < +\infty$ удовлетворяют равенству:

$$x_{i+n} = a_n x_i + a_{n-1}x_{i-1} + \ldots + a_2 x_{i+n-2} + a_1 x_{i+n-1} = \sum_{j=1}^{n} a_j\, x_{i+n-j} ,$$

где все операции выполнены в поле GF(2). Эта формула называется законом рекурсии, который генерирует последовательность, а вектор (x_0, x_1 ,…, x_{n-1}) – начальным вектором (зародышем, состоянием) последовательности.

Выходная последовательность LFSR единственным образом определяется многочленом обратной связи и начальным заполнением регистра. Для LFSR длины n с коэффициентами a_1, a_2 , ,..., a_n.

многочлен обратной связи: $P(x) = 1 - \sum_{i=1}^{n} a_i x^i$

Альтернативно для задания выходной последовательности LFSR можно использовать:

характеристический многочлен: $P^*(x) = x^n P(1/x) - \sum_{i=1}^{n} a_i x^{n-i}$

2.3 Конструкция прореживающего генератора

В [8] конструкция прореживающего генератора использует два источника псевдослучайных битов, чтобы создать третий источник псевдослучайных битов (потенциально) лучшего качества, чем исходные источники. Здесь качество означает трудность предсказания псевдослучайной последовательности Последовательность, которая создается на выходе, является подпоследовательностью из первого источника, где элементы подпоследовательности выбираются в соответствии с позициями битов "1" во втором источнике. Другими словами, пусть a_0 , a_1 ,... обозначает первую последовательность, а s_0 , s_1 ,... - вторую. Мы создаем третью последовательность x_0, x_1,..., которая включает в себя те биты a, для которых соответствующий s_i равен ‘1’. Остальные биты из первой последовательности отбрасываются. (Таким образом, результирующая последовательность является “сокращенной" версией первой). Формально, для всех k = 0,l..., $x_k = a_{i_k}$, где x_k - позиция k-го ‘1’ в последовательности s_0 , s_1 ,... Назовем результирующий генератор псевдослучайных чисел прореживающим генератором (shrinking generator (SG)).

Эта общая идея может быть применена к любой паре псевдослучайных источников. Здесь анализируется конструкция, в которой два источника генерируются с использованием регистров сдвига с линейной обратной связью (LFSR). LFSR - это очень хорошо известные структуры, состоящие из сдвигового регистра, управляемого тактовым сигналом, который при каждом тактовом импульсе выводит свой старший бит, сдвигает его содержимое в наиболее значимом направлении и вводит бит в его менее значимую позицию. Этот бит обратной связи вычисляется как линейная комбинация (через GF(2)) битов в сдвиговом регистре. Эта линейная комбинация может быть фиксированной (например, подключенной в аппаратной реализации) или переменной. В последнем случае линейная комбинация (или соединение)

определяется двоичным вектором длины LFSR. (В аппаратной реализации это достигается с помощью дополнения к регистру сдвига, программируемого управляющего регистра, который определяет ячейки регистра сдвига, подключенные к схеме исключающего ИЛИ (XOR)).

Обозначим через A первый LFSR в нашей конструкции, а через S (для управляющего) - второй. |A| и |S| - обозначают их длины, а последовательности, которые они генерируют (после фиксации соединений и начального содержимого регистров), обозначаются как $a_0, a_1,...,$ и $s_0, s_1,...,$ соответственно. Наконец, результирующая сокращенная последовательность, обозначаемая этой конструкцией, хорошо определена как для фиксированных, так и для переменных соединений LFSR. В случае реализации фиксированного соединения, только начальные значения (т.е. начальное содержимое регистров сдвига) для LFSR A и S представляют собой секретный ключ для генератора псевдослучайных значений (или ключ шифрования/дешифрования, когда он используется в качестве потокового шифра). Если используются переменные (программируемые) соединения, то значение этих соединений также является частью ключа. Для прореживающего генератора задействованы LFSR, которые являются независимыми, например, их периоды являются взаимно простыми.

Свойства SG-генератора выражаются следующими двумя утверждениями [6]:

Теорема 1. Пусть T_a, T_s – соответственно периоды последовательностей $\{a_t\}$, $\{s_t\}$. Если генераторы G_1, G_2 используют примитивные порождающие многочлены степеней L_1 и L_2 соответственно, а периоды T_a, T_s – взаимно простые числа, то выходная последовательность $\{x_t\}$ имеет период

$$T = (2^{L_1} - 1)\, 2^{L_2 - 1}$$

Теорема 2. В условиях предыдущей теоремы линейная сложность выходной последовательности $\{x_t\}$ удовлетворяет неравенствам

$$L_1 2^{L_2 - 2} \le L\{x_t\} \le L_1 2^{L_2 - 1}$$

2.4 Монобит-тест для генераторов

В криптографии и статистике важно проверять случайность и равномерность распределения выходных последовательностей генераторов псевдослучайных чисел (ГПСЧ). Для анализа свойств прореживающего и самосжимающего генераторов применяются различные статистические тесты, к примеру, монобит-тест.

Монобит-тест — это один из базовых статистических тестов, применяемых для оценки случайности битовых последовательностей, генерируемых псевдослучайными генераторами. Он входит в состав

стандартных пакетов тестов, таких как NIST SP 800-22, и служит для первичной оценки равномерности распределения нулей и единиц в последовательности.

Суть монобит-теста заключается в проверке того, достаточно ли сбалансированы количества нулей и единиц в сгенерированной последовательности. Для идеального случайного генератора ожидается, что число нулей и единиц в длинной последовательности будет приблизительно одинаковым, без значимого перекоса в сторону одной из величин. Если такой перекос есть, это может указывать на систематические отклонения, ошибки в алгоритме генерации или утрату свойств случайности.

Применение этого теста особенно актуально при анализе последовательностей, полученных от **самосжимающих** и **прореживающих генераторов**:

- **Самосжимающие генераторы** формируют выходную последовательность, удаляя часть исходных битов в зависимости от внутреннего состояния или по определённому правилу. Это может изменить статистические характеристики последовательности, включая соотношение нулей и единиц.
- **Прореживающие генераторы** также формируют выходные данные за счёт отбора отдельных битов (например, каждый второй, третий и т.д.) или в зависимости от значений других битов, что также влияет на равномерность распределения.

Монобит-тест позволяет на первом этапе обнаружить нарушения равномерности, вызванные особенностями сжатия или прореживания. Если генератор существенно смещает частоту появления одного из битов, это будет зафиксировано на этом этапе, что служит основанием для дальнейшего, более глубокого анализа.

2.5 Период последовательности прореживающего генератора

В [11] используется экспоненциальная оценка периода последовательностей, создаваемых прореживающим генератором. В случае периода эта оценка является жесткой. Важность длинного периода заключается в том, чтобы избежать повторения последовательности через короткие промежутки времени.

Пусть A и S образуют прореживающий генератор X. Введем обозначение T_A, T_S - периоды A- и S- последовательностей соответственно.

Если:

- A и S имеют максимальную длину (т.е. имеют примитивные соединения)

- $(T_A, T_S) = 1$, тогда последовательность X имеет период $T_A \times 2^{|S|-1} = (2^{|A|} - 1) \times 2^{|S|-1}$

Примечание: S не должна быть максимальной длины. В общем случае период последовательности X-последовательности - это $T_A \times$Ws, где Ws - число единиц в полном периоде S. Если оба периода T_A , T_S имеют максимальную длину, тогда условие $((T_A, T_S) = 1$ эквивалентно $(|T_A|, |T_S|)= 1$.

Тогда формула периода последовательности прореживающего генератора задается формулой:

$$T_{shr}= T_S \times (2^{|L1|} - 1),$$

где:

$T_A = 2^{|L1|} - 1$ - период последовательности LFSR1 с примитивным многочленом степени L1.

$T_S=2^{|L2|-1}$ - период управляющего LFSR2 длины L2,

$\gcd(T_A, T_S)=1$ (взаимная простота периодов).

За один полный цикл LFSR1 генерирует T_A битов, из которых примерно $\frac{T_A}{2}$ единиц (по свойству равномерного распределения битов в LFSR). Для формирования выходной последовательности требуется $T_A \times T_S$ тактов, чтобы LFSR1 и LFSR2 прошли полные циклы, однако из-за прореживания выходная последовательность формируется только в те моменты, когда LFSR1 выдает 1. Поскольку $\gcd(T_A, T_S)=1$, комбинация состояний LFSR1 и LFSR2 повторяется через $T_A \times T_S$ тактов, но выходная последовательность сокращается из-за отбрасывания битов.

За время T_A LFSR2 совершит $\frac{T_A}{\gcd(T_A,T_S)} = T_A$ циклов и выходная последовательность состоит из $\frac{T_A \times T_S}{T_A} = T_S$ битов, отобранных за каждый цикл LFSR1.

2.5 Конструкция самосжимающего генератора

В [11] показано, что самосжимающий генератор может быть применен к произвольным двоичным последовательностям. Исходная последовательность a $=(a_0, a_1, a_2.)$ рассматривается как последовательность пар битов $((a_0, a_1),(a_2, a_3),...)$. Если пара (a_{2i}, a_{2i+1}) равна значению (1, 0) или

(1, 1), то используется псевдослучайный бит 0 или 1 соответственно. С другой стороны, если пара равна (0,0) или (0, 1), она будет отброшена, что означает, что она не будет вносить выходной бит в новую последовательность s = (s_0, s_1, s_2..).

Самосжимающий генератор, в частности, предназначен для применения к псевдослучайным последовательностям с целью получения новых псевдослучайных последовательностей (потенциально) лучшего криптографического качества, поэтому специально анализируется ситуация, когда исходная последовательность a генерируется с помощью LFSR. Для криптографического приложения ключ состоит из начального состояния LFSR. Предпочтительно, чтобы обратная связь была переменной и также являлась частью ключа. Самосжимающую последовательность s можно рассматривать как полученную из исходной последовательности a путем отбрасывания определенных бит. Ожидается, что в среднем 3/4 бит будут пропущены. Следовательно, скорость передачи данных исходной последовательности уменьшается в 4 раза.

Чтобы показать, что самосжимающий генератор может быть реализован как частный случай сжимающегося генератора возьмем a = (a_0, a_1, a_2..) - последовательность, полученная с помощью LFSR длины N, определяющей Самосжимающий генератор. Согласно правилу самосжимания, последовательность (a_0 , a_2 , a_4 ..) влияет на выходное управление, а (a_1 , a_3 , a_5 ,..) определяет управляемую последовательность. Обе последовательности могут быть созданы исходным LFSR при загрузке с исходными состояниями (a_0, a_2, a_{2N-2}..) или (a_1, a_3, a_{2N-1}.)) соответственно. Это означает, что самосжимающий генератор может быть реализован как прореживающий генератор с двумя SRs-генераторами, имеющими идентичные соединения обратной связи [11].

Рассмотрим генератор произвольного сжатия, определяемый двумя линейными сдвиговыми регистрами LFSR 1 и LFSR 2 с полиномами обратной связи f(x) и g(x), соответственно. Кроме того, пусть b =(b_0, b_1,b_2,..) и c = (c_0 ,c_1 , c_2 ..)) обозначают соответствующие выходные последовательности LFSR. Затем, применяя правило самосжимания к чередующейся последовательности a = (c_0 , b_0 , c_1, b_1 , ..) получаем исходную выходную последовательность прореживающего генератора. С другой стороны, можно показать, что последовательность a может быть получена с помощью LFSR с полиномом обратной связи f(x^2)g(x^2) = f(x)^2 g(x)^2 . Это означает, что прореживающий генератор имеет эквивалентную реализацию в качестве самосжимающего генератора.

2.6 Период последовательности самосжимающего генератора

Пусть a = $(a_0, a_1, a_2..)$ - выходная последовательность нетривиально инициализированного m-LFSR длины L. Следовательно, a - это последовательность с периодом $2^L - 1$. Самосжимающая последовательность также будет периодической. Фактически, после $2\,(2^L - 1)$ битов исходной последовательности, последовательность пар (a_0, a_1), (a_2, a_3), ..., (a_{2^L-2}, a_0), (a_1, a_2), ..., (a_{2^L-3}, a_{2^L-2}) обработано, и следующей парой снова будет (a_0, a_1).

Следовательно, сокращенная последовательность повторяется. В течение этого периода каждая возможная выходная пара (a_i , a_{i+1}), $0 \leq$ i $<2^L - 1$, из исходной LFSR-последовательности выходит ровно один раз. Как известно, в течение периода m-LFSR-последовательности каждая из пар 01, 10 и 11 появляется ровно 2^{L-2} раза, а пара 00 появляется $2^{L-2} - 1$ раз. Из определения правила сокращения следует, что 2^{L-1} – это период сокращенной последовательности. Более того, поскольку пары 10 и 11 встречаются одинаково часто, сокращенная последовательность должна быть сбалансирована. Поскольку сокращенная последовательность повторяется через 2^{L-1} бит, она должна быть чисто периодической с периодом p = 2^{L-1}, т.е. $s_n = s_{n+p}$ для всех n > 0. Это означает, что наименьший период P из s должен делить 2^{L-1}. Подводя итог, мы получаем:

В [6]: *Теорема 1*. Пусть a - M-LFSR-последовательность, порожденная LFSR длиной L, и пусть s – самосжимающая последовательность, полученная из a. Тогда s – сбалансированная последовательность, период которой делится на 2^{L-1}.

Нижняя граница периода сокращенной M-LFSR-последовательности приведена в следующей теореме.

Теорема 2. Период P самосжимающей LFSR-последовательности максимальной длины, созданной LFSR длиной L, удовлетворяет неравенству P $\geq 2^{[L/2]}$.

2.7 М-последовательность

В [9] в качестве примера построим генератор М-последовательностей для порождающего полинома j (x)=x^4A x^3A

1. Для рассматриваемого примера имеем

$$\begin{bmatrix} a_1(k+1) \\ a_2(k+1) \\ a_3(k+1) \\ a_4(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \bullet \begin{bmatrix} a_1(k) \\ a_2(k) \\ a_3(k) \\ a_4(k) \end{bmatrix}.$$

Для построения одноканального генератора М-последовательности получим систему логических уравнений:

$$a_1(k+1) = a_3(k) \oplus a_4(k)$$

$$a_2(k+1) = a_1(k)$$

$$a_1(k+1) = a_2(k)$$

$$a_1(k+1) = a_3(k)$$

На рис.1.3 изображена схема генератора М-последовательности для полинома в соответствии с системой логических уравнений. Регистр сдвига реализован на D-триггерах, состояния которых изменяются по приходу на С-входы тактовых импульсов.

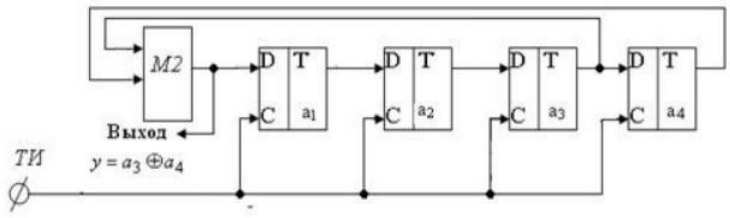


Рис.1.3

№ такта a_1 a_2 a_3 a_4 Выход

Длина формируемой последовательности равна 15, т.е. через 15 тактов регистр устанавливается в начальное состояние.

Перейдём к рассмотрению свойств последовательностей максимальной длины:

1. Период М-последовательности, формируемой в соответствии с выражением $a_k = \sum_{i=1}^{m} a_i a_{k-i}$, k = 0, 1, 2, 3, …,

где $a_k \in \{0,1\}$ – символы последовательности; $a_i \in \{0, 1\}$- коэффициенты, определяемые примитивным неприводимым порождающим полиномом φ(x), для которого m=deg φ(x), равен 2^m -1

2. Для заданного φ(x) существует L различных М-последовательностей, сдвинутых относительно друг друга.

3. Число единичных символов на периоде М-последовательности равно 2^{m-1} , а нулевых - 2^{m-1} -1. Вероятности появления 1 и 0 определяются выражениями: $p(a_i = 1) = \frac{2^{m-1}}{2^m - 1} = \frac{1}{2} + \frac{1}{2^{m+1} - 2}$;

$$p(a_i = 0) = \frac{2^{m-1} - 1}{2^m - 1} = \frac{1}{2} - \frac{1}{2^{m+1} - 2}$$

и при увеличении m достигают значений сколь угодно близких к 0,5.

4. В псевдослучайной последовательности максимальной длины серии из одного символа (1 или 0) встречаются 2^{m-2} раз, из двух единиц или нулей 2^{m-3} и т.д. Серии из m-1 нулей и m единиц встречаются лишь по одному разу. Сравнивая выражения для оценки вероятности появлении серий из I одинаковых символов, можно убедиться в их практической эквивалентности.

5. Для каждого целого s(I≤s< L) существует такое целое r ≠ s(I≤s< L), что $\{a_i\}$A$\{a_i$-s$\}=\{a_i$-r$\}$. Данное свойство обычно называют свойством сдвига и сложения.

6. Автокорреляционная функция М-последовательности определяется выражением:

$$R_a(\text{Ţ}) = \begin{cases} 1 - npur = 0(modL), \\ -\frac{1}{L} - npur \neq 0(modL). \end{cases}$$

Децимацией последовательности $\{a_i\}$ по индексу q (q=1, 2, 3, …) называется формирование новой последовательности {bi} из q—х элементов $\{a_i\}$, т.е. b_k=a_{kq}. Если $\{b_i\}$ является нулевой последовательностью, то она порождается полиномом ψ (x), и имеет период L/(L, q), где (L, q)наибольший общий делитель L и q. При (L, q) = 1 период $\{b_i\}$ равен L=2^m-1, где m=deg φ (x), и децимация называется собственной или нормальной.

Результатом всякой нормальной децимации является М–последовательность периода L, порождаемая примитивным неприводимым полиномом ψ (x). Децимация выполняется над последовательностью, сдвинутой на j тактов относительно исходной $\{a_i\}$, то получаемая последовательность будет также сдвинутой на некоторое число j тактов по сравнению с $\{bi\}$. Иначе говоря, независимо от того, какой именно сдвиг последовательности, порождаемой полиномом φ (x), выбран, результатом ее всегда оказывается М–последовательность, порождаемая полиномом ψ (x). В частности, при децимации характеристической М—последовательности $\{a_i\}$*, порождаемой многочленом φ (x), получается также характеристическая последовательность $\{b_i\}$*, соответствующая полиному ψ (x).

3. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

3.1 Тестирование прореживающего генератора

Реализуем монобит-тест с целью демонстрации того, что выходные последовательности являются равномерно распределёнными:

Вывод:

s_obs = 1.02

p_value = 0.307,

где s_obs – статистика наблюдаемого отклонения.

Из вывода следует, что последовательность проходит монобит-тест, что дает четкое представление об его корректной работе и позволяет использовать данный генератор на практике.

Также данный тест был применен к следующей группе полиномов:

P1(x)=$x^7 + x^4 + x^3 + x^2 + 1$ и P2(x)=$x^{11} + x^{10} + x^8 + x + 1$.

P1(x)= $x^{25} + x^{23} + x^{15} + x^{13} + x^{10} + x^7 + 1$ и P2(x)= $x^{18} + x^{10} + x^8 + x^7 + x^4 + x + 1$

P1(x)= $x^{62} + x^{48} + x^{18} + x^{16} + x^9 + x^2 + 1$ и P2(x)= $x^{13} + x^{12} + x^{10} + x^5 + x^2 + x + 1$

P1(x)=$x^{41} + x^{32} + x^{31} + x^{27} + 1$ и P2(x)=$x^{20} + x^{13} + x^7 + x^2 + 1$.

P1(x)=$x^{41} + x^{32} + x^{31} + x^{27} + 1$ и P2(x)=$x^{39} + x^{35} + x^{23} + x^{16} + 1$.

Все полиномы успешно прошли данный тест.

Далее возьмем пару полиномов для вычисления периода:

P1(x)=$x^{47} + x^{32} + x^{24} + + x^{11} + 1$ и P2(x)=$x^{17} + x^{16} + x^{12} + x^4 + 1$.

Длина L1 = 47, длина L2 = 17.

Условия:

1. Проверка примитивности полиномов (необходима факторизация $2^{|L1|}$ - 1):
   - Для L=47: $2^{47}-1$=2351·4513·13264529·... (составное).
   - Для L= 2^{17-1}= $2^{16} = 65536$(составное)

Если полиномы примитивны, их периоды равны T1=$2^{47}-1$*, T2=*2^{17-1}.

2. Взаимная простота периодов:

gcd($2^{47}-1$, 2^{17-1})= $2^{gsd(47,16)}-1=2^1-1=1$.

Период выходной последовательности:

T_{shr}=(2^{17-1})·($2^{47}-1$) ≈ 9.2233·10^{18}

Таким образом были получены теоретические результаты вычисления периода последовательностей прореживающего генератора для данных степеней, которые сходятся с практическими результатами, что подтверждает корректность компьютерной реализации. Также были проведены эксперименты с другими полиномами.

3.2 Реализация прореживающего генератора

Теперь сгенерируем последовательность, которая получается в результате следующего алгоритма:

1. LFSR-генератор с порождающим многочленом степени L_1 порождает «элементарную» двоичную последовательность $\{a_t\}$;

2. LFSR-генератор с порождающим многочленом степени L_2 порождает двоичную «управляющую» последовательность $\{s_t\}$;

3. С помощью этих двух последовательностей $\{a_t\}$, $\{s_t\}$ строится выходная последовательность $\{x_t\}$, которая включает те биты a_t, для которых соответствующее значение $s_t = 1$; если $s_t = 0$, то значение a_t игнорируется.

Для работы возьмем примитивные полиномы вида P1(x)=$x^7 + x + 1$ и P2(x)=$x^{11} + x^2 + 1$ (приложение Д).

Получаем выходную последовательность, сгенерированную прореживающим генератором (приложение Б).

Далее задаем некоторое L, берем начальный фрагмент длины L^2 из последовательности и формируем в виде матрицы, после чего вычисляется ее ранг над полем GF(2).

В результате чего для k=1,…,1024 получилась следующая выходная последовательность, состоящая из рангов матриц:

[1, 1, 2, 2, 4, 5, 7, 8, 8, 10, 11, 11, 12, 14, 13, 15, 15, 17, 18, 19, 21, 21, 23, 24, 24, 26, 27, 27, 28, 28, 30, 31, 32, 33, 34, 35, 36, 36, 38, 40, 41, 42, 42, 43, 44, 46, 47, 47, 48, 49, 51, 51, 51, 54, 54, 55, 56, 56, 58, 59, 60, 62, 60, 63, 65, 65, 66, 67, 69, 70, 69, 72, 72, 74, 75, 76, 76, 77, 78, 80, 81, 81, 82, 84, 85, 84, 85, 87, 89, 90, 90, 92, 92, 93, 92, 94, 95, 98, 99, 99, 101, 101, 101, 103, 104, 106, 106, 107, 108, 109, 110, 111, 112, 114, 113, 116, 116, 116, 117, 119, 120, 122, 121, 123, 124, 126, 125, 56, 129, 129, 130, 131, 132, 134, 134, 135, 136, 137, 137, 139, 140, 141, 143, 143, 145, 144, 147, 148, 149, 150, 151, 152, 152, 153, 155, 155, 155, 157, 158, 159, 161, 161, 163, 163, 164, 166, 166, 167, 168, 169, 170, 172, 172, 174, 173, 175, 176, 176, 178, 180, 181, 182, 182, 184, 184, 186, 185, 188, 189, 189, 191, 112, 193, 194, 193, 195, 196, 198, 199, 200, 200, 200, 203, 203, 204, 206, 206, 208, 208, 208, 209, 211, 212, 212, 214, 216, 217, 218, 218, 220, 221, 221, 221, 224, 225, 225, 226, 227, 228, 230, 231, 231, 233, 233, 234, 235, 236, 238, 238, 240, 239, 239, 243, 243, 245, 246, 246, 246, 249, 250, 250, 251, 253, 253, 254, 28, 255, 257, 258, 259, 261, 261, 262, 263, 263, 266, 265, 267, 269, 268, 270, 271, 273, 274, 275, 275, 276, 278, 279, 279, 279, 279, 282, 282, 285, 285, 287, 224, 289, 289, 289, 291, 292, 293, 295, 295, 296, 298, 297, 298, 300, 300, 302, 303, 304, 305, 307, 307, 307, 309, 310, 311, 313, 314, 315, 315, 316, 316, 317, 112, 321, 321, 322, 323, 324, 326, 326, 327, 328, 329, 330, 331, 333, 333, 334, 335, 336, 338, 338, 340, 339, 341, 341, 343, 343, 345, 345, 348, 347, 349, 349, 224, 351, 353, 355, 356, 355, 358, 359, 360, 360, 360, 362, 363, 365, 366, 366, 367, 368, 370, 371, 371, 372, 373, 374, 375, 377, 376, 379, 379, 381, 381, 382, 56, 383, 385, 387, 385, 389, 390, 389, 392, 393, 393, 394, 395, 396, 398, 399, 400, 400, 402, 403, 402, 404, 405, 405, 407, 408, 409, 410, 411, 412, 413, 413, 224, 416, 417, 418, 419, 421, 421, 422, 424, 424, 426, 427, 427, 428, 428, 430, 431, 432, 433, 434, 435, 437, 437, 439, 440, 439, 442, 442, 444, 444, 445, 446, 112, 449, 450, 451, 451, 452, 454, 455, 455, 456, 458, 458, 459, 460, 461, 463, 448, 465, 466, 466, 467, 467, 469, 470, 472, 472, 473, 474, 475, 477, 477, 478, 224, 480, 481, 482, 483, 484, 485, 487, 487, 488, 490, 490, 492, 493, 493, 495, 448, 496, 498, 498, 498, 500, 501, 502, 502, 503, 506, 506, 256, 509, 509, 510, 14, 513, 513, 514, 516, 516, 516, 518, 519, 520, 522, 523, 523, 525, 524, 527, 448, 527, 529, 530, 530, 532, 532, 535, 535, 536, 538, 539, 539, 540, 539, 542, 224, 545, 545, 546, 547, 548, 549, 549, 551, 552, 553, 555, 555, 557, 557, 558, 448, 560, 561, 561, 564, 564, 565, 566, 568, 568, 569, 571, 570, 573, 574, 573, 112, 577, 577, 578, 580, 580, 581, 583, 584, 584, 584, 586, 587, 588, 590, 591, 448, 592, 593, 594, 596, 596, 598, 599, 599, 600, 601, 603, 603, 604, 605, 606, 224, 609, 610, 610, 612, 612, 614, 614, 615, 616, 618, 619, 619, 620, 621, 623, 448, 625, 625, 626, 628, 628, 628, 630, 631, 633, 632, 635, 636, 637, 637, 639, 56, 640, 641, 642, 643, 645, 645, 646, 646, 649, 650, 650, 652, 652, 653, 654, 448, 656, 658, 659, 659, 661, 661, 662, 663, 664, 664, 666, 667, 668, 669, 670, 224, 673, 674, 674, 675, 676, 676, 678, 679, 680, 681, 682, 683, 684, 685, 685, 448, 687, 689, 690, 690, 692, 693, 694, 695, 697, 698, 696, 699, 701, 701, 702, 112, 705, 705, 706, 707, 709, 709, 710, 712, 711, 714, 715, 714, 717, 717, 718, 448, 720, 721, 721, 722, 724, 726, 727, 727, 729, 728, 730, 732, 732, 734, 733, 224, 734, 737, 738, 739, 740, 741, 742, 743, 745, 745, 746, 746, 748, 749, 749, 448, 753, 753, 754, 755, 757, 757, 759, 760, 760, 512, 763, 763, 765, 765, 767, 28, 769, 769, 770, 770, 772, 774, 774, 776, 776, 777, 779, 779, 780, 781, 781, 448, 785, 784, 786, 787, 789, 788, 790, 792, 791, 792, 794, 795, 796, 798, 7

97, 224, 800, 799, 803, 804, 805, 805, 807, 808, 809, 809, 811, 810, 813, 813, 814, 448, 817, 817, 817, 820, 820, 82
2, 822, 823, 823, 825, 827, 827, 829, 830, 831, 112, 833, 834, 834, 835, 836, 837, 838, 839, 840, 841, 841, 843, 844,
846, 845, 448, 848, 849, 851, 850, 852, 853, 852, 856, 857, 857, 858, 860, 860, 862, 863, 224, 864, 865, 866, 867, 8
68, 868, 869, 871, 873, 872, 874, 876, 876, 877, 877, 448, 880, 881, 882, 882, 884, 885, 887, 887, 889, 890, 891, 89
2, 893, 893, 894, 56, 896, 898, 898, 899, 901, 901, 902, 896, 904, 905, 906, 907, 908, 910, 910, 448, 912, 913, 913,
916, 916, 917, 919, 896, 919, 922, 922, 923, 924, 926, 926, 224, 929, 929, 930, 932, 932, 933, 934, 896, 937, 937, 9
38, 939, 941, 942, 943, 448, 945, 944, 946, 947, 948, 949, 950, 896, 952, 954, 953, 955, 956, 957, 958, 112, 960, 96
1, 963, 963, 965, 965, 965, 896, 967, 969, 971, 971, 972, 973, 974, 448, 976, 977, 977, 979, 980, 981, 982, 896, 985,
985, 987, 987, 988, 990, 990, 224, 992, 994, 995, 996, 996, 997, 997, 896, 1001, 1001, 1003, 1002, 1004, 1006, 100
7, 448, 1007, 1009, 1011, 1011, 1011, 1014, 1014, 128, 1017, 1018, 1018, 1018, 1020, 1020, 1022, 7]

Зависимость порядка матрицы от ее ранга представлена на (рис.1.1 приложения Г).

Теперь рассмотрим примитивные полиномы вида P1(x)=$x^7 + x^4 + x^3 +$ $+x^2 + 1$ и P2(x)=$x^{11} + x^{10} + x^8 + x + 1$.

Для k=1,…,1024 получилась следующая выходная последовательность, состоящая из рангов матриц:

[1, 1, 2, 4, 5, 6, 7, 8, 9, 9, 9, 12, 12, 14, 15, 15, 16, 17, 19, 20, 20, 22, 20, 24, 24, 26, 25, 26, 28, 29, 30, 31, 32, 33, 3
5, 36, 37, 37, 38, 39, 40, 41, 42, 42, 44, 46, 47, 47, 49, 49, 49, 51, 53, 53, 54, 56, 56, 57, 57, 59, 60, 61, 63, 63, 64, 6
5, 67, 68, 67, 69, 70, 71, 72, 72, 74, 75, 77, 76, 77, 80, 81, 82, 82, 83, 84, 85, 85, 88, 88, 89, 91, 91, 91, 93, 94, 96, 9
6, 97, 97, 99, 101, 100, 102, 104, 105, 106, 107, 108, 109, 109, 110, 112, 112, 113, 115, 114, 117, 117, 117, 119, 11
9, 121, 122, 123, 124, 126, 126, 56, 128, 129, 130, 131, 131, 133, 134, 135, 136, 137, 137, 139, 140, 142, 142, 143,
144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 155, 155, 156, 157, 158, 160, 160, 162, 163, 162, 163, 165, 167, 1
66, 168, 169, 170, 170, 172, 174, 175, 174, 176, 178, 179, 180, 180, 181, 180, 183, 184, 186, 186, 187, 188, 189, 18
9, 112, 191, 193, 195, 194, 197, 198, 198, 199, 200, 201, 203, 202, 204, 205, 206, 207, 209, 209, 210, 211, 213, 213,
214, 216, 217, 217, 219, 219, 221, 221, 222, 224, 225, 225, 226, 227, 229, 228, 230, 231, 233, 233, 234, 236, 236, 2
38, 239, 239, 239, 242, 243, 243, 243, 244, 246, 247, 248, 249, 251, 251, 252, 253, 254, 28, 256, 257, 258, 259, 261,
261, 262, 263, 265, 266, 267, 267, 268, 269, 271, 270, 271, 274, 274, 275, 275, 277, 278, 279, 281, 281, 281, 283, 2
83, 285, 287, 224, 288, 289, 290, 291, 293, 293, 294, 294, 296, 297, 298, 299, 300, 302, 302, 303, 303, 305, 306, 30
7, 308, 309, 310, 310, 312, 313, 314, 316, 316, 317, 318, 112, 320, 321, 322, 323, 324, 325, 326, 326, 328, 329, 331,
331, 331, 333, 335, 333, 336, 337, 338, 338, 340, 340, 342, 343, 344, 345, 347, 347, 348, 348, 351, 224, 352, 353, 3
55, 355, 355, 357, 358, 359, 361, 362, 362, 363, 364, 364, 366, 367, 368, 370, 370, 371, 373, 373, 375, 375, 376, 37
7, 378, 378, 380, 382, 383, 56, 383, 385, 387, 388, 387, 389, 390, 392, 392, 393, 395, 395, 396, 398, 399, 399, 400,
402, 403, 404, 405, 405, 406, 408, 408, 409, 411, 411, 412, 413, 413, 224, 415, 418, 417, 419, 420, 422, 422, 424, 4
24, 426, 426, 428, 429, 429, 431, 430, 432, 433, 435, 435, 436, 436, 438, 440, 440, 441, 442, 443, 443, 445, 447, 11
2, 449, 450, 450, 450, 453, 453, 454, 455, 456, 457, 456, 458, 460, 461, 462, 448, 464, 465, 466, 468, 469, 469, 469,
471, 472, 472, 475, 476, 475, 477, 478, 224, 480, 481, 483, 483, 485, 486, 486, 488, 489, 489, 491, 491, 492, 493, 4
94, 448, 495, 497, 498, 498, 499, 501, 502, 503, 505, 505, 506, 256, 509, 509, 510, 14, 512, 514, 515, 515, 516, 517,
519, 519, 520, 521, 523, 522, 524, 526, 526, 448, 529, 529, 530, 531, 533, 533, 535, 535, 537, 537, 538, 539, 540, 5
41, 542, 224, 544, 545, 546, 547, 548, 549, 549, 552, 553, 553, 554, 555, 557, 557, 558, 448, 561, 559, 562, 561, 56
4, 565, 565, 568, 569, 569, 571, 572, 572, 572, 574, 112, 576, 577, 578, 579, 581, 581, 582, 584, 584, 585, 586, 588,
587, 589, 590, 448, 592, 593, 594, 595, 596, 596, 597, 599, 601, 601, 602, 603, 604, 606, 606, 224, 608, 610, 610, 6
11, 612, 613, 615, 615, 616, 618, 618, 619, 620, 620, 622, 448, 624, 625, 627, 627, 629, 630, 629, 631, 633, 633, 63
4, 635, 636, 637, 639, 56, 640, 641, 643, 643, 645, 645, 646, 648, 648, 649, 651, 652, 652, 654, 654, 448, 657, 657,
659, 659, 660, 662, 662, 664, 664, 665, 667, 668, 667, 670, 670, 224, 672, 673, 673, 676, 675, 678, 679, 680, 680, 6
82, 682, 684, 685, 686, 686, 448, 689, 689, 690, 692, 692, 693, 694, 695, 696, 698, 698, 699, 700, 700, 702, 112, 70
4, 705, 706, 708, 709, 710, 710, 711, 710, 714, 714, 716, 715, 718, 717, 448, 720, 722, 723, 723, 724, 725, 726, 727,
729, 730, 730, 731, 732, 733, 734, 224, 737, 737, 739, 739, 741, 741, 743, 742, 743, 746, 746, 748, 747, 748, 750, 4
48, 752, 753, 754, 756, 756, 757, 758, 760, 761, 512, 761, 763, 765, 766, 766, 28, 768, 768, 770, 772, 772, 772, 774,
776, 777, 776, 779, 779, 780, 782, 782, 448, 784, 784, 786, 787, 789, 790, 791, 791, 792, 793, 794, 795, 796, 798, 7
98, 224, 800, 800, 802, 802, 803, 805, 807, 807, 809, 809, 811, 812, 812, 812, 814, 448, 816, 817, 819, 819, 820, 82
1, 823, 823, 823, 824, 826, 827, 829, 829, 830, 112, 833, 834, 834, 836, 837, 838, 839, 839, 840, 840, 842, 843, 844,
846, 846, 448, 848, 850, 851, 852, 853, 853, 854, 856, 855, 857, 859, 860, 860, 862, 862, 224, 865, 865, 867, 867, 8
68, 869, 871, 871, 872, 874, 873, 875, 875, 876, 879, 448, 880, 881, 882, 883, 885, 885, 886, 887, 888, 889, 891, 89
1, 892, 892, 894, 56, 897, 896, 899, 899, 900, 901, 903, 896, 905, 905, 906, 908, 907, 908, 910, 448, 913, 913, 914,
915, 917, 918, 917, 896, 921, 921, 922, 922, 924, 925, 926, 224, 928, 930, 929, 932, 932, 932, 935, 896, 936, 938, 9
38, 940, 940, 941, 943, 448, 945, 945, 946, 948, 948, 949, 950, 896, 953, 953, 954, 955, 956, 957, 959, 112, 960, 96
0, 962, 964, 964, 964, 966, 896, 968, 968, 970, 972, 973, 973, 974, 448, 976, 977, 978, 978, 980, 981, 983, 896, 984,

984, 987, 987, 989, 989, 990, 224, 992, 992, 993, 994, 996, 998, 999, 896, 1000, 1001, 1001, 1003, 1004, 1006, 100
7, 448, 1008, 1009, 1010, 1012, 1012, 1014, 1014, 128, 1016, 1017, 1018, 1019, 1020, 1021, 1022, 7]

Зависимость порядка матрицы от ее ранга представлена на (рис.1.2 приложения Г).

И для примитивных полиномов вида P1(x)= $x^7 + x^5 + x^4 + x^3 + x^2 +$ $+x + 1$ и P2(x)=$x^{11} + x^9 + x^7 + x^5 + x^2 + x + 1$.

Для k=1,…,1024 получилась следующая выходная последовательность, состоящая из рангов матриц:

[1, 1, 2, 3, 4, 5, 7, 7, 8, 8, 10, 12, 12, 13, 15, 15, 17, 17, 18, 17, 20, 20, 22, 22, 24, 25, 26, 27, 28, 28, 30, 32, 32, 34,
34, 35, 37, 37, 39, 39, 40, 42, 42, 43, 44, 45, 45, 47, 47, 48, 50, 51, 52, 53, 54, 55, 54, 56, 59, 60, 61, 61, 62, 63, 64,
66, 65, 67, 67, 70, 69, 72, 72, 74, 73, 76, 76, 77, 79, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95,
96, 98, 99, 100, 99, 101, 101, 103, 104, 105, 106, 108, 108, 109, 109, 112, 112, 113, 114, 116, 116, 117, 118, 119, 1
20, 121, 123, 123, 125, 126, 126, 56, 128, 128, 129, 131, 132, 132, 134, 135, 136, 136, 139, 139, 141, 141, 142, 143,
144, 145, 146, 148, 148, 149, 150, 151, 151, 153, 155, 156, 156, 157, 159, 160, 159, 161, 162, 163, 164, 164, 166, 1
68, 168, 168, 171, 171, 172, 174, 173, 175, 176, 177, 179, 178, 180, 181, 183, 183, 184, 186, 187, 187, 189, 188, 19
0, 112, 191, 193, 193, 196, 197, 198, 199, 199, 200, 202, 203, 203, 203, 205, 206, 207, 208, 209, 209, 211, 212, 213,
215, 215, 217, 217, 217, 219, 221, 221, 222, 224, 224, 225, 226, 228, 228, 229, 230, 231, 232, 233, 233, 234, 236, 2
37, 239, 239, 239, 241, 242, 243, 244, 245, 247, 247, 248, 249, 250, 252, 252, 254, 254, 28, 256, 257, 258, 260, 261,
262, 263, 263, 264, 265, 266, 267, 267, 268, 270, 270, 273, 274, 274, 276, 276, 278, 278, 279, 280, 282, 282, 284, 2
85, 285, 286, 224, 288, 290, 290, 292, 291, 294, 294, 295, 296, 296, 299, 299, 300, 300, 302, 303, 303, 306, 306, 30
8, 308, 309, 309, 311, 312, 314, 314, 316, 315, 317, 317, 112, 320, 321, 321, 323, 324, 326, 324, 327, 329, 329, 330,
331, 333, 333, 335, 336, 335, 337, 338, 340, 340, 342, 341, 343, 344, 346, 347, 348, 347, 349, 349, 224, 352, 354, 3
54, 355, 356, 357, 358, 359, 359, 362, 363, 363, 363, 365, 367, 367, 368, 370, 370, 372, 372, 374, 374, 376, 376, 37
7, 378, 380, 380, 380, 381, 56, 384, 384, 386, 388, 388, 388, 391, 391, 391, 394, 395, 395, 397, 397, 398, 399, 399,
402, 402, 403, 404, 406, 406, 407, 409, 410, 411, 412, 411, 413, 414, 224, 417, 417, 418, 419, 421, 421, 422, 423, 4
24, 425, 426, 427, 429, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 440, 439, 441, 441, 443, 444, 445, 447, 11
2, 449, 448, 449, 451, 452, 453, 454, 456, 455, 457, 458, 459, 461, 461, 463, 448, 464, 466, 466, 467, 468, 469, 470,
472, 472, 474, 474, 474, 477, 478, 478, 224, 480, 480, 482, 482, 484, 485, 485, 487, 489, 489, 489, 491, 492, 493, 4
95, 448, 496, 498, 498, 500, 501, 502, 500, 503, 503, 505, 507, 256, 507, 509, 510, 14, 513, 513, 514, 515, 515, 518,
518, 520, 520, 521, 521, 523, 525, 526, 526, 448, 528, 530, 530, 531, 532, 533, 534, 536, 536, 538, 539, 539, 540, 5
42, 542, 224, 544, 545, 546, 547, 548, 549, 550, 552, 553, 553, 554, 556, 557, 557, 558, 448, 561, 562, 562, 563, 56
5, 565, 567, 568, 568, 570, 569, 571, 572, 573, 574, 112, 577, 578, 578, 580, 581, 581, 583, 583, 585, 585, 586, 587,
588, 588, 591, 448, 592, 593, 595, 595, 596, 598, 598, 599, 601, 600, 602, 603, 604, 605, 606, 224, 607, 610, 610, 6
10, 612, 612, 614, 614, 616, 616, 619, 620, 620, 621, 622, 448, 625, 626, 626, 626, 628, 629, 630, 631, 632, 634, 63
4, 635, 636, 637, 638, 56, 640, 641, 641, 644, 644, 644, 646, 648, 647, 649, 651, 652, 652, 653, 654, 448, 657, 658,
658, 659, 660, 662, 661, 663, 665, 665, 666, 668, 668, 669, 670, 224, 672, 674, 674, 675, 675, 677, 678, 680, 681, 6
81, 682, 683, 684, 686, 685, 448, 689, 689, 690, 690, 692, 694, 694, 695, 696, 697, 698, 700, 699, 700, 703, 112, 70
4, 705, 706, 708, 709, 710, 710, 711, 712, 713, 713, 715, 716, 717, 719, 448, 721, 721, 723, 724, 724, 726, 727, 727,
729, 729, 730, 732, 731, 733, 735, 224, 736, 737, 739, 739, 741, 741, 742, 743, 745, 746, 747, 746, 748, 749, 751, 4
48, 752, 752, 754, 755, 755, 758, 759, 759, 759, 512, 763, 764, 765, 766, 766, 28, 769, 769, 770, 771, 771, 773, 774,
775, 777, 777, 778, 780, 781, 782, 782, 448, 784, 785, 787, 787, 789, 789, 790, 791, 791, 793, 793, 795, 797, 797, 7
99, 224, 801, 802, 803, 804, 804, 805, 807, 807, 808, 808, 810, 812, 812, 813, 814, 448, 816, 817, 819, 819, 820, 82
1, 820, 824, 824, 826, 826, 827, 829, 829, 830, 112, 832, 834, 833, 836, 836, 837, 838, 838, 840, 842, 841, 844, 845,
845, 846, 448, 848, 850, 851, 852, 853, 854, 854, 855, 856, 858, 858, 859, 860, 861, 863, 224, 865, 866, 866, 868, 8
68, 870, 871, 870, 873, 874, 874, 876, 876, 877, 878, 448, 879, 881, 883, 883, 884, 885, 887, 887, 888, 889, 891, 89
1, 893, 893, 894, 56, 896, 898, 898, 900, 901, 900, 902, 896, 904, 906, 906, 907, 908, 909, 911, 448, 912, 913, 914,
915, 916, 917, 919, 896, 920, 921, 922, 923, 924, 926, 926, 224, 928, 929, 930, 929, 932, 933, 934, 896, 936, 936, 9
38, 939, 940, 941, 941, 448, 944, 945, 946, 948, 948, 949, 951, 896, 951, 954, 955, 955, 956, 957, 958, 112, 960, 96
1, 963, 962, 964, 964, 966, 896, 969, 969, 970, 971, 972, 973, 975, 448, 976, 977, 977, 979, 979, 981, 983, 896, 985,
984, 987, 988, 988, 990, 989, 224, 992, 994, 993, 995, 997, 998, 998, 896, 1001, 1001, 1002, 1003, 1005, 1005, 100
7, 448, 1008, 1010, 1011, 1011, 1013, 1014, 1014, 128, 1017, 1017, 1019, 1020, 1019, 1022, 1021, 7]

Зависимость порядка матрицы от ее ранга представлена на (рис.1.3 приложения Г).

Анализ предоставленных выходных последовательностей и их графических отображений указывает на наличие существенных корреляций

между последовательностями рангов матриц над полем GF(2) для примитивных многочленов заданных степеней. Несмотря на отсутствие полной идентичности последовательностей, графики демонстрируют схожие паттерны падений рангов. В точках этих падений ранги матриц либо совпадают, либо находятся в непосредственной близости друг от друга, различаясь на незначительные величины порядка 1-2 единиц.

Аналогичные эксперименты были проведены для следующих примитивных полиномов P1(x)= $x^{25}+x^{23}+x^{15}+x^{13}+x^{10}+x^{7}+1$ и P2(x)=$x^{18}+x^{10}+x^{8}+x^{7}+x^{4}+x+1$ (рис.1.4 приложение Г).

Для P1(x)=$x^{62}+x^{48}+x^{18}+x^{16}+x^{9}+x^{2}+1$ и P2(x)=$x^{13}+x^{12}+x^{10}+x^{5}+x^{2}+x+1$ (рис.1.5 приложение Г).

Для P1(x)=$x^{63}+x^{60}+x^{22}+x^{18}+x^{8}+x^{5}+1$ и P2(x)=$x^{11}+x^{9}+x^{7}+x^{5}+x^{2}+x+1$ (рис.1.6 приложение Г).

Также были рассмотрены следующие пары порождающих многочленов:

P1(x)=$x^{47}+x^{32}+x^{24}+x^{11}+1$ и P2(x)=$x^{17}+x^{16}+x^{12}+x^{4}+1$.

P1(x)=$x^{41}+x^{32}+x^{31}+x^{27}+1$ и P2(x)=$x^{20}+x^{13}+x^{7}+x^{2}+1$.

P1(x)=$x^{41}+x^{32}+x^{31}+x^{27}+1$ и P2(x)=$x^{39}+x^{35}+x^{23}+x^{16}+1$.

Введем следующие обозначения:

- L_1 - степень порождающего многочлена порождающего (элементарного) регистра сдвига;
- L_2 - степень порождающего многочлена управляющего регистра сдвига;
- l – порядок матрицы;
- m – уровень или итерация,

тогда зависимость может быть представлена следующим образом:

$$R = 2^{L_2-m}k+2^{L_2-m-1}$$

где R – ранг матрицы.

Таблица 1 зависимости:

Ранг R	Порядок l
L_1	$2^{L_2-1}k$
$2L_1$	$2^{L_2-1}k+2^{L_2-2}$
$4L_1$	$2^{L_2-2}k+2^{L_2-3}$
…	…
$2^{m}L_1$	$2^{L_2-m}k+2^{L_2-m-1}$
…	…
$2^{L_2-1}L_1$	$2k+1$ (все нечётные)

- Здесь k∈ N∪{0} и $\begin{cases} k \geq 2^{2m-L_2}L_1 - \frac{1}{2}, m > 0; \\ k \geq 2^{1-L_2}. m = 0. \end{cases}$.
- Условие необходимо, чтобы значение ранга не превосходило значение порядка.

Данная зависимость четко прослеживается в том случае, когда степень второго многочлена мала, соответственно, чтобы избежать этого, необходимо брать степень управляющего многочлена достаточно большой.

Если степени двух примитивных многочленов приблизительно равны, то таким образом возможно задать период какой-то определенной длины, потому что если L_1 – большая, а L_2 - маленькая, то будет прослеживаться данная зависимость.

Период на этих экспериментах еще не был достигнут, а зависимость уже существует, что представляет практический интерес.

Зависимость показывает, что ранг матрицы увеличивается с увеличением как порядка матрицы k, так и уровня итерации m; степени порождающих многочленов (L_1 и L_2) играют важную роль в определении конечного ранга матрицы; анализ рангов для различных k и m позволяет выявлять закономерности и аномалии в последовательностях, генерируемых LFSR.

3.3 Тестирование самосжимающего генератора

Для проверки равномерного распределения выходной последовательности самосжимающего генератора, реализуем монобит-тест:

Вывод:

S_obs: 0.018

p-value: 0.9856

где s_obs – статистика наблюдаемого отклонения.

Из вывода следует, что последовательность проходит монобит-тест, что дает представление об его корректной работе и позволяет использовать данный генератор на практике.

Также данный тест был применен к следующей группе полиномов:

P(x)=$x^{14} + x^{12} + x^{11} + x + 1$, P(x)=$x^{18} + x^{16} + x^{11} + +x^4 + 1$.

P(x)=$x^{41} + x^{32} + x^{31} + x^{27} + 1$, P(x)=$x^{39} + x^{35} + x^{23} + x^{16} + 1$.

P(x)=$x^7 + x^4 + x^3 + x^2 + 1$, P(x)=$x^{13} + x^8 + x^5 + x^3 + 1$.

P(x)=$x^{24} + x^{11} + x^5 + x^2 + 1$, P(x)=$x^{25} + x^{13} + x^{12} + x^7 + 1$.

Все полиномы успешно прошли данный тест.

Теперь вычислим период для полинома : P(x)=$x^{11} + x^{10} + x^8 + +x + 1$, длина L = 11.

T_{ssr}=$2^{11-1} = 2^{10}$= 1024

Результат:

```
Период выходной последовательности: 1024
```

Таким образом были получены теоретические результаты вычисления периода последовательностей самосжимающего генератора для малых степеней, которые сходятся с практическими результатами, что подтверждает корректность компьютерной реализации. Также были проведены эксперименты с другими полиномами.

3.4 Реализация самосжимающего генератора

Теперь сгенерируем последовательность, которая получается в результате следующего алгоритма:

1. **Инициализация регистра сдвига**:

Установить начальное состояние регистра сдвига (LFSR) длины L, который формирует последовательность на основе порождающего многочлена степени L;

2. **Обработка битов по парам**:

Сдвинуть регистр на два такта так, чтобы получить пару битов: (b1,b2), где b1 - первый бит в паре, а b2 – второй;

3. **Условие выбора бита для выхода**:

- Если b1=1, то b2 включается в выходную последовательность,
- Если b1=0, то оба бита (b1, b2) игнорируются;

4. **Повторение сдвига**:

- Если пара битов была отброшена (b1=0), выполнить ещё два такта для получения новой пары (b3, b4),
- Продолжать обработку пар в соответствии с шагами 2-3.

5. **Завершение процесса**:

Продолжать генерацию и обработку битов до тех пор, пока не будет достигнута заданная длина выходной последовательности;

Самосжимающий генератор отличается от прореживающего тем, что обе последовательности (исходная и управляющая) формируются из одного регистра, что упрощает его реализацию.

Для работы возьмем примитивный полином вида P(x)=$x^{11} + x^{10} + x^8 + +x + 1$ (приложение Д).

Получаем выходную последовательность, сгенерированную самосжимающим генератором (приложение Б).

Далее задаем некоторое L, берем начальный фрагмент длины L^2 из последовательности и формируем в виде матрицы, после чего вычисляется ее ранг над полем GF(2).

В результате чего для k=1,…,1024 получилась следующая выходная последовательность, состоящая из рангов матриц:

[1, 1, 3, 4, 5, 4, 6, 7, 9, 9, 10, 11, 12, 14, 13, 14, 16, 18, 18, 19, 20, 22, 23, 24, 23, 26, 26, 27, 28, 29, 30, 32, 33, 33, 33, 35, 37, 38, 37, 40, 40, 41, 42, 43, 45, 46, 46, 46, 49, 50, 50, 51, 52, 53, 54, 54, 56, 58, 58, 59, 60, 62, 62, 16, 64, 65, 66, 68, 69, 68, 70, 71, 72, 73, 74, 76, 75, 77, 78, 64, 81, 82, 82, 83, 85, 85, 86, 86, 88, 90, 90, 91, 92, 93, 94, 32, 95, 96, 99, 99, 100, 101, 102, 103, 104, 105, 106, 106, 107, 109, 110, 64, 112, 113, 114, 116, 116, 117, 118, 118, 120, 122, 123, 124, 123, 126, 127, 8, 128, 129, 131, 131, 132, 134, 135, 127, 135, 138, 138, 139, 141, 141, 142, 64, 144, 145, 147, 147, 149, 149, 150, 127, 153, 153, 155, 155, 157, 157, 158, 32, 161, 160, 162, 163, 164, 165, 166, 127, 168, 168, 170, 171, 172, 173, 174, 64, 177, 177, 179, 179, 180, 182, 183, 127, 184, 185, 186, 187, 188, 189, 189, 16, 192, 193, 195, 195, 197, 195, 199, 127, 201, 201, 203, 203, 205, 205, 205, 64, 208, 209, 211, 211, 211, 212, 214, 127, 215, 217, 218, 219, 220, 221, 223, 32, 224, 226, 227, 228, 228, 229, 230, 127, 232, 233, 234, 235, 236, 237, 239, 64, 240, 241, 243, 243, 244, 245, 246, 127, 247, 249, 250, 251, 252, 253, 254, 4, 256, 257, 259, 254, 260, 260, 262, 127, 264, 265, 267, 254, 267, 269, 270, 64, 271, 272, 273, 254, 276, 278, 277, 127, 280, 281, 281, 254, 283, 286, 286, 32, 288, 289, 290, 254, 292, 294, 295, 127, 296, 297, 298, 254, 299, 301, 303, 64, 304, 305, 306, 254, 308, 310, 310, 127, 311, 314, 314, 254, 316, 317, 318, 16, 321, 321, 322, 254, 324, 325, 327, 127, 329, 330, 330, 254, 333, 332, 334, 64, 336, 338, 339, 254, 341, 342, 342, 127, 344, 345, 347, 254, 348, 349, 350, 32, 352, 354, 354, 254, 357, 356, 358, 127, 361, 362, 362, 254, 365, 365, 366, 64, 367, 370, 370, 254, 373, 373, 375, 127, 376, 377, 379, 254, 380, 381, 382, 8, 384, 385, 386, 254, 388, 390, 390, 127, 391, 393, 395, 254, 396, 398, 398, 64, 400, 401, 402, 254, 404, 406, 406, 127, 408, 409, 411, 254, 413, 414, 414, 32, 416, 417, 419, 254, 421, 422, 422, 127, 425, 425, 425, 254, 429, 430, 429, 64, 432, 433, 435, 254, 437, 437, 438, 127, 441, 440, 441, 254, 444, 445, 447, 16, 448, 448, 451, 254, 453, 453, 455, 127, 456, 457, 458, 254, 460, 461, 462, 64, 463, 465, 466, 254, 468, 469, 470, 127, 471, 472, 474, 254, 474, 477, 477, 32, 480, 481, 482, 254, 484, 485, 486, 127, 488, 489, 490, 254, 492, 493, 495, 64, 496, 497, 498, 254, 500, 501, 502, 127, 504, 506, 506, 254, 508, 507, 508, 2, 512, 507, 514, 254, 516, 507, 517, 127, 521, 507, 522, 254, 525, 507, 526, 64, 528, 507, 529, 254, 531, 507, 534, 127, 537, 507, 538, 254, 541, 507, 543, 32, 545, 507, 546, 254, 547, 507, 550, 127, 551, 507, 553, 254, 556, 507, 559, 64, 559, 507, 563, 254, 564, 507, 566, 127, 569, 507, 570, 254, 572, 507, 574, 16, 577, 507, 579, 254, 580, 507, 582, 127, 584, 507, 586, 254, 588, 507, 590, 64, 593, 507, 595, 254, 597, 507, 598, 127, 600, 507, 602, 254, 604, 507, 606, 32, 608, 507, 610, 254, 613, 507, 614, 127, 616, 507, 617, 254, 619, 507, 623, 64, 624, 507, 626, 254, 629, 507, 630, 127, 632, 507, 635, 254, 634, 507, 638, 8, 640, 507, 642, 254, 644, 507, 646, 127, 648, 507, 650, 254, 652, 507, 654, 64, 656, 507, 658, 254, 660, 507, 663, 127, 665, 507, 666, 254, 669, 507, 669, 32, 673, 507, 674, 254, 676, 507, 679, 127, 679, 507, 681, 254, 684, 507, 686, 64, 688, 507, 691, 254, 691, 507, 695, 127, 696, 507, 698, 254, 700, 507, 702, 16, 704, 507, 707, 254, 708, 507, 711, 127, 712, 507, 715, 254, 716, 507, 719, 64, 720, 507, 722, 254, 724, 507, 727, 127, 728, 507, 729, 254, 732, 507, 734, 32, 735, 507, 738, 254, 740, 507, 743, 127, 745, 507, 747, 254, 748, 507, 750, 64, 751, 507, 754, 254, 757, 507, 759, 127, 760, 507, 761, 254, 764, 507, 766, 4, 768, 507, 770, 254, 771, 507, 775, 127, 776, 507, 778, 254, 780, 507, 782, 64, 784, 507, 785, 254, 788, 507, 791, 127, 792, 507, 795, 254, 797, 507, 799, 32, 800, 507, 802, 254, 803, 507, 807, 127, 807, 507, 810, 254, 812, 507, 814, 64, 816, 507, 818, 254, 821, 507, 822, 127, 825, 507, 826, 254, 828, 507, 830, 16, 832, 507, 834, 254, 837, 507, 838, 127, 840, 507, 842, 254, 843, 507, 845, 64, 848, 507, 850, 254, 852, 507, 855, 127, 856, 507, 858, 254, 860, 507, 863, 32, 864, 507, 866, 254, 869, 507, 871, 127, 872, 507, 874, 254, 876, 507, 878, 64, 880, 507, 882, 254, 884, 507, 886, 127, 888, 507, 891, 254, 892, 507, 895, 8, 896, 507, 898, 254, 901, 507, 902, 127, 904, 507, 906, 254, 908, 507, 911, 64, 912, 507, 914, 254, 916, 507, 918, 127, 919, 507, 922, 254, 923, 507, 926, 32, 928, 507, 930, 254, 932, 507, 934, 127, 936, 507, 938, 254, 941, 507, 941, 64, 944, 507, 946, 254, 949, 507, 950, 127, 953, 507, 953, 254, 956, 507, 958, 16, 960, 507, 962, 254, 964, 507, 967, 127, 968, 507, 970, 254, 972, 507, 974, 64, 975, 507, 977, 254, 980, 507, 982, 127, 984, 507, 987, 254, 988, 507, 990, 32, 991, 507, 994, 254, 996, 507, 999]

Зависимость порядка матрицы от ее ранга представлена на (рис.2.1 приложения Г).

Теперь рассмотрим примитивный полином вида P(x)=$x^{14} + x^{12} + x^{11} + +x + 1$.

Для k=1,…,1024 получилась следующая выходная последовательность, состоящая из рангов матриц:

[1, 1, 2, 4, 4, 5, 7, 7, 8, 9, 11, 12, 11, 14, 15, 15, 16, 17, 19, 19, 20, 21, 23, 23, 24, 23, 25, 28, 28, 30, 30, 31, 33, 33, 34, 35, 36, 38, 38, 39, 39, 42, 42, 43, 44, 46, 46, 47, 49, 48, 50, 51, 52, 54, 54, 55, 56, 57, 58, 59, 61, 62, 62, 63, 64, 66, 66, 68, 68, 70, 71, 72, 72, 74, 74, 75, 76, 76, 78, 79, 80, 82, 82, 82, 83, 85, 85, 88, 88, 89, 90, 91, 91, 93, 94, 96, 95, 96, 98, 99, 101, 101, 102, 103, 103, 104, 106, 108, 108, 110, 110, 111, 112, 113, 114, 115, 115, 117, 118, 119, 121, 122, 121, 124, 124, 125, 126, 64, 129, 130, 131, 131, 133, 133, 134, 134, 135, 137, 139, 139, 141, 141, 142, 143, 144, 145, 146, 147, 147, 149, 150, 150, 152, 153, 155, 155, 156, 157, 158, 159, 161, 162, 163, 163, 165, 165, 166, 167, 168, 168, 169, 171, 172, 173, 174, 175, 176, 178, 178, 179, 181, 181, 182, 183, 185, 185, 185, 187, 187, 189, 190, 128, 192, 193, 194, 195, 197, 198, 198, 199, 201, 201, 203, 203, 204, 205, 206, 206, 208, 209, 210, 211, 213, 212, 214, 215, 216, 217, 219, 219, 220, 221, 222, 223, 224, 226, 226, 227, 229, 230, 230, 232, 233, 232, 235, 235, 236, 238, 237, 240, 240, 240, 243, 243, 244, 246, 247, 248, 247, 250, 251, 251, 252, 253, 254, 32, 254, 258, 257, 259, 261, 261, 262, 263, 263, 264, 267, 267, 268, 269, 270, 271, 272, 273, 273, 276, 276, 276, 278, 279, 280, 280, 283, 283, 2

85, 285, 285, 256, 287, 289, 290, 291, 292, 292, 294, 295, 297, 297, 299, 300, 300, 301, 303, 302, 304, 304, 307, 30
8, 308, 310, 311, 311, 312, 312, 315, 316, 317, 316, 317, 128, 321, 321, 322, 323, 323, 325, 326, 327, 327, 329, 331,
331, 333, 333, 335, 336, 336, 337, 337, 339, 340, 341, 342, 343, 344, 345, 346, 346, 348, 350, 350, 256, 352, 353, 3
53, 353, 356, 357, 357, 359, 361, 361, 362, 363, 363, 365, 367, 368, 368, 370, 369, 371, 372, 373, 373, 375, 376, 37
7, 379, 380, 381, 380, 382, 64, 385, 385, 386, 387, 388, 390, 390, 391, 392, 392, 394, 395, 397, 398, 398, 400, 400,
402, 403, 404, 404, 406, 406, 407, 408, 408, 410, 411, 412, 413, 414, 256, 417, 417, 419, 419, 419, 421, 422, 423, 4
25, 424, 427, 427, 429, 430, 431, 431, 433, 433, 435, 435, 436, 438, 439, 439, 441, 440, 443, 443, 444, 446, 447, 12
8, 449, 448, 451, 451, 452, 452, 454, 455, 456, 457, 458, 459, 460, 460, 461, 463, 464, 465, 466, 468, 468, 468, 470,
471, 472, 473, 474, 476, 477, 477, 479, 256, 481, 481, 483, 483, 484, 486, 486, 486, 488, 488, 490, 491, 493, 494, 4
94, 495, 496, 496, 498, 499, 500, 501, 503, 503, 504, 505, 507, 507, 507, 510, 509, 16, 513, 514, 514, 515, 516, 517,
518, 519, 520, 522, 522, 524, 524, 526, 525, 512, 528, 529, 530, 532, 532, 534, 534, 535, 537, 537, 538, 539, 540, 5
42, 541, 256, 545, 546, 547, 546, 549, 549, 551, 551, 552, 554, 555, 555, 556, 557, 559, 512, 560, 561, 563, 563, 56
5, 565, 566, 567, 568, 569, 570, 571, 572, 574, 574, 128, 575, 577, 579, 579, 581, 581, 582, 584, 583, 585, 585, 587,
589, 589, 591, 512, 592, 593, 594, 595, 597, 598, 599, 600, 601, 601, 601, 604, 604, 605, 607, 256, 608, 609, 611, 6
11, 613, 613, 614, 615, 616, 618, 618, 620, 620, 621, 623, 512, 624, 625, 626, 627, 628, 628, 630, 631, 632, 633, 63
5, 636, 637, 637, 638, 64, 640, 641, 641, 643, 644, 645, 645, 648, 648, 648, 650, 650, 652, 653, 654, 512, 656, 657,
657, 660, 659, 661, 662, 664, 665, 664, 667, 668, 668, 670, 670, 256, 671, 674, 674, 675, 676, 677, 678, 679, 681, 6
82, 682, 683, 685, 686, 687, 512, 689, 689, 690, 691, 692, 693, 694, 696, 696, 698, 698, 698, 699, 701, 702, 128, 70
4, 705, 705, 708, 709, 708, 710, 711, 712, 713, 714, 715, 715, 718, 717, 512, 719, 721, 723, 724, 724, 724, 726, 727,
728, 729, 730, 730, 731, 734, 734, 256, 736, 737, 739, 739, 741, 741, 741, 743, 743, 745, 746, 747, 747, 750, 750, 5
12, 752, 752, 754, 755, 755, 757, 758, 758, 760, 762, 762, 763, 764, 765, 766, 32, 768, 770, 770, 772, 772, 773, 774,
775, 776, 778, 778, 780, 780, 781, 783, 512, 784, 785, 786, 787, 788, 789, 790, 792, 792, 793, 795, 796, 797, 798, 7
98, 256, 801, 799, 803, 803, 805, 804, 805, 808, 808, 809, 810, 811, 812, 812, 815, 512, 815, 817, 818, 820, 821, 82
1, 822, 823, 824, 825, 826, 828, 827, 829, 830, 128, 833, 834, 834, 834, 836, 838, 838, 839, 841, 842, 842, 843, 844,
845, 846, 512, 848, 849, 849, 851, 852, 854, 854, 854, 857, 857, 858, 859, 859, 860, 862, 256, 864, 865, 866, 867, 8
68, 870, 870, 871, 872, 873, 875, 876, 876, 877, 878, 512, 880, 881, 883, 883, 883, 886, 887, 887, 888, 890, 891, 89
2, 891, 894, 894, 64, 896, 897, 898, 899, 901, 902, 902, 904, 904, 905, 907, 907, 907, 909, 910, 512, 912, 912, 915,
915, 916, 917, 918, 919, 920, 921, 921, 923, 924, 925, 926, 256, 928, 929, 930, 931, 932, 933, 934, 936, 936, 938, 9
38, 939, 941, 941, 942, 512, 943, 945, 946, 946, 948, 949, 950, 952, 952, 954, 954, 955, 955, 956, 958, 128, 960, 96
1, 963, 964, 964, 966, 966, 968, 968, 969, 969, 971, 973, 973, 974, 512, 976, 978, 979, 980, 981, 982, 981, 982, 985,
985, 987, 987, 988, 990, 990, 256, 992, 993, 995, 996, 996, 996, 999]

Зависимость порядка матрицы от ее ранга представлена на (рис.2.2 приложения Г).

Теперь рассмотрим примитивный полином вида P(x)=$x^{18} + x^{16} + x^{11} + +x^4 + 1$.

Для k=1,…,1024 получилась следующая выходная последовательность, состоящая из рангов матриц:

[1, 1, 1, 3, 4, 5, 7, 8, 8, 9, 10, 12, 13, 14, 15, 16, 17, 17, 19, 18, 21, 20, 22, 22, 24, 26, 25, 27, 29, 29, 30, 31, 32, 34,
34, 35, 37, 38, 39, 38, 39, 41, 42, 44, 44, 46, 47, 47, 48, 49, 51, 51, 52, 53, 54, 56, 56, 58, 59, 60, 61, 61, 62, 63, 65,
65, 65, 68, 69, 69, 70, 71, 73, 74, 74, 75, 76, 77, 78, 79, 80, 81, 82, 84, 83, 85, 86, 87, 87, 89, 90, 91, 92, 94, 94, 96,
97, 96, 97, 100, 101, 100, 103, 102, 104, 105, 107, 108, 108, 109, 110, 112, 113, 113, 115, 115, 116, 117, 118, 119,
121, 121, 123, 124, 125, 124, 125, 127, 128, 128, 130, 132, 133, 134, 135, 134, 136, 136, 138, 140, 139, 142, 142, 1
43, 143, 145, 146, 148, 148, 149, 151, 150, 152, 153, 154, 156, 157, 158, 159, 159, 159, 161, 163, 163, 165, 165, 16
6, 167, 168, 170, 170, 172, 172, 173, 175, 175, 176, 178, 179, 179, 179, 181, 182, 183, 184, 185, 187, 187, 189, 189,
190, 192, 193, 192, 194, 195, 197, 197, 197, 198, 201, 202, 202, 203, 204, 204, 206, 207, 209, 210, 211, 210, 213, 2
12, 214, 215, 216, 217, 218, 219, 221, 221, 221, 224, 224, 225, 226, 227, 227, 229, 230, 231, 231, 233, 235, 236, 23
6, 238, 237, 240, 239, 241, 243, 244, 244, 246, 246, 247, 248, 249, 250, 250, 251, 252, 254, 255, 257, 257, 258, 260,
261, 262, 262, 264, 265, 265, 266, 268, 267, 269, 270, 271, 271, 274, 274, 275, 277, 278, 278, 279, 281, 281, 282, 2
83, 284, 285, 286, 286, 289, 290, 291, 291, 292, 293, 294, 295, 296, 296, 298, 299, 301, 301, 302, 304, 304, 305, 30
6, 308, 308, 310, 310, 311, 313, 313, 315, 315, 315, 317, 319, 319, 321, 321, 322, 323, 324, 324, 327, 328, 327, 329,
330, 331, 331, 334, 335, 336, 335, 337, 338, 340, 340, 340, 342, 343, 344, 345, 347, 346, 348, 348, 350, 351, 353, 3
53, 354, 355, 357, 358, 359, 358, 360, 361, 363, 363, 365, 365, 366, 368, 368, 368, 371, 371, 372, 373, 373, 376, 37
6, 378, 379, 379, 381, 381, 382, 382, 383, 385, 386, 387, 388, 388, 390, 392, 392, 393, 394, 396, 396, 397, 399, 400,
400, 401, 402, 403, 404, 405, 406, 405, 408, 410, 410, 411, 411, 413, 413, 416, 417, 416, 418, 420, 421, 421, 422, 4
23, 424, 425, 427, 427, 429, 429, 429, 431, 433, 433, 434, 435, 435, 436, 438, 439, 441, 440, 443, 443, 444, 444, 44
6, 446, 449, 449, 450, 450, 451, 453, 454, 455, 455, 457, 459, 459, 459, 461, 462, 462, 465, 465, 466, 467, 468, 469,
471, 471, 473, 473, 473, 476, 476, 476, 478, 479, 480, 481, 483, 483, 484, 486, 485, 487, 489, 489, 491, 491, 492, 4
93, 494, 496, 496, 498, 498, 499, 501, 501, 502, 503, 504, 505, 506, 506, 508, 510, 511, 256, 510, 514, 514, 515, 51

7, 517, 518, 519, 519, 522, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 536, 536, 537, 538, 539, 541, 542, 542, 544, 544, 545, 547, 547, 549, 549, 550, 551, 553, 553, 555, 555, 557, 557, 558, 559, 559, 562, 562, 563, 565, 565, 565, 567, 569, 569, 570, 571, 571, 574, 574, 575, 577, 576, 578, 579, 580, 581, 582, 584, 584, 585, 585, 586, 588, 590, 591, 591, 592, 593, 594, 595, 597, 597, 598, 599, 600, 602, 602, 604, 604, 604, 607, 607, 609, 608, 611, 610, 612, 613, 614, 615, 617, 617, 619, 619, 621, 622, 622, 624, 624, 626, 627, 628, 628, 629, 630, 630, 630, 634, 634, 634, 636, 638, 638, 639, 640, 641, 641, 643, 644, 645, 647, 647, 648, 649, 650, 651, 652, 654, 654, 656, 655, 657, 658, 659, 660, 661, 662, 664, 665, 666, 666, 667, 669, 669, 671, 672, 673, 673, 675, 676, 677, 678, 678, 679, 680, 681, 683, 684, 684, 685, 686, 688, 687, 689, 690, 692, 692, 693, 694, 695, 696, 698, 698, 699, 701, 701, 702, 703, 705, 706, 707, 707, 709, 710, 710, 712, 712, 713, 714, 715, 717, 717, 718, 719, 720, 721, 723, 723, 724, 725, 726, 727, 729, 729, 730, 732, 732, 732, 734, 735, 736, 738, 738, 740, 740, 742, 742, 743, 744, 745, 747, 748, 748, 749, 750, 752, 752, 753, 754, 755, 755, 758, 758, 760, 760, 762, 763, 764, 765, 766, 766, 512, 769, 769, 770, 771, 772, 774, 773, 774, 776, 777, 779, 779, 781, 781, 783, 783, 784, 784, 785, 787, 788, 789, 790, 792, 792, 793, 794, 796, 796, 796, 797, 799, 800, 801, 803, 803, 804, 804, 806, 806, 808, 808, 810, 812, 812, 814, 814, 815, 816, 817, 817, 820, 820, 821, 822, 822, 823, 825, 827, 827, 828, 828, 830, 832, 831, 834, 835, 836, 837, 837, 838, 840, 840, 842, 842, 842, 844, 846, 846, 848, 849, 849, 850, 852, 852, 853, 854, 854, 856, 858, 858, 859, 860, 861, 862, 863, 864, 864, 866, 867, 868, 870, 871, 871, 872, 873, 874, 875, 875, 878, 878, 879, 880, 882, 882, 883, 884, 885, 886, 886, 888, 890, 890, 891, 892, 894, 894, 893, 897, 897, 898, 900, 901, 902, 903, 903, 904, 905, 906, 906, 907, 910, 911, 912, 912, 914, 914, 915, 916, 917, 919, 918, 921, 921, 922, 923, 925, 925, 926, 928, 928, 930, 930, 931, 932, 932, 935, 935, 935, 937, 938, 939, 938, 940, 941, 943, 945, 946, 947, 947, 948, 949, 951, 951, 952, 952, 955, 955, 956, 958, 958, 960, 959, 961, 961, 964, 965, 964, 966, 968, 968, 970, 970, 971, 973, 973, 974, 976, 976, 976, 979, 980, 979, 981, 982, 984, 984, 985, 986, 986, 989, 990, 990, 991, 991, 993, 995, 994, 997, 998, 998]

Зависимость порядка матрицы от ее ранга представлена на (рис.2.3 приложения Г).

Теперь рассмотрим примитивный полином вида P(x)=$x^{63} + x^{54} + x^{44} + x^{20} + 1$.

Для k=1,…,1024 получилась следующая выходная последовательность, состоящая из рангов матриц:

[1, 1, 1, 1, 1, 2, 4, 6, 7, 8, 9, 10, 12, 13, 14, 16, 17, 18, 17, 20, 21, 21, 22, 22, 24, 25, 26, 28, 29, 29, 30, 30, 32, 33, 33, 35, 37, 37, 38, 39, 41, 41, 42, 43, 43, 45, 47, 47, 49, 50, 50, 50, 52, 51, 54, 55, 56, 58, 58, 59, 59, 61, 62, 63, 64, 65, 66, 67, 69, 68, 70, 72, 71, 73, 74, 75, 77, 77, 78, 80, 80, 81, 82, 83, 84, 85, 86, 87, 87, 88, 91, 92, 92, 94, 94, 95, 96, 98, 98, 99, 100, 102, 103, 103, 104, 104, 106, 107, 108, 109, 110, 111, 113, 113, 114, 115, 116, 117, 118, 118, 121, 122, 122, 123, 124, 125, 126, 126, 127, 129, 129, 131, 132, 133, 134, 135, 136, 137, 139, 140, 139, 141, 142, 143, 143, 145, 147, 148, 148, 148, 150, 152, 152, 153, 153, 155, 156, 157, 157, 159, 161, 161, 161, 164, 165, 165, 167, 167, 168, 169, 171, 171, 172, 174, 174, 175, 176, 177, 178, 178, 180, 182, 183, 183, 185, 185, 187, 187, 188, 190, 190, 191, 192, 194, 194, 195, 197, 196, 197, 199, 201, 202, 203, 202, 205, 206, 206, 206, 209, 209, 210, 210, 212, 212, 214, 214, 216, 217, 218, 220, 220, 222, 222, 223, 223, 224, 227, 228, 227, 228, 231, 230, 233, 232, 235, 234, 236, 237, 238, 239, 240, 241, 242, 244, 244, 244, 247, 248, 249, 249, 250, 251, 252, 254, 254, 255, 256, 257, 259, 259, 260, 261, 261, 263, 265, 265, 265, 267, 268, 269, 271, 271, 272, 274, 274, 275, 277, 277, 278, 280, 281, 281, 283, 283, 284, 285, 286, 288, 288, 289, 290, 291, 292, 293, 294, 294, 296, 297, 298, 300, 300, 302, 302, 304, 305, 306, 306, 307, 309, 308, 311, 311, 312, 314, 314, 315, 316, 317, 318, 319, 321, 322, 322, 323, 324, 325, 327, 328, 328, 330, 331, 332, 332, 333, 334, 336, 336, 336, 339, 340, 339, 341, 343, 342, 345, 346, 347, 347, 348, 349, 350, 351, 353, 352, 355, 356, 357, 357, 358, 359, 361, 361, 362, 363, 365, 365, 367, 367, 368, 370, 370, 371, 372, 373, 374, 374, 377, 378, 378, 379, 380, 381, 382, 384, 384, 385, 386, 388, 388, 388, 389, 392, 392, 392, 395, 395, 395, 398, 398, 398, 401, 402, 403, 403, 405, 405, 406, 406, 409, 408, 411, 411, 412, 413, 414, 415, 416, 418, 418, 419, 420, 422, 423, 423, 424, 425, 426, 427, 429, 429, 430, 432, 432, 434, 435, 435, 436, 438, 439, 439, 441, 442, 441, 443, 444, 445, 447, 447, 448, 450, 450, 451, 452, 453, 454, 455, 456, 457, 458, 458, 460, 461, 462, 464, 465, 465, 467, 468, 467, 470, 470, 471, 472, 474, 474, 475, 476, 477, 478, 479, 480, 481, 482, 484, 484, 484, 486, 487, 489, 489, 491, 491, 492, 493, 493, 495, 496, 497, 497, 500, 500, 501, 502, 503, 504, 505, 507, 506, 509, 510, 510, 511, 512, 512, 514, 514, 517, 517, 518, 518, 519, 521, 522, 524, 525, 525, 527, 527, 529, 530, 530, 531, 533, 533, 534, 535, 536, 537, 539, 539, 541, 540, 542, 544, 545, 545, 547, 547, 549, 550, 550, 550, 552, 553, 554, 556, 556, 558, 557, 560, 560, 561, 562, 563, 565, 565, 566, 567, 566, 568, 570, 571, 572, 572, 575, 575, 576, 578, 578, 580, 580, 581, 582, 584, 585, 586, 585, 588, 589, 590, 591, 592, 592, 593, 595, 595, 596, 598, 597, 600, 601, 600, 603, 602, 604, 605, 606, 608, 609, 610, 610, 611, 612, 613, 614, 615, 616, 618, 617, 619, 620, 622, 622, 624, 623, 625, 627, 626, 629, 629, 630, 631, 632, 633, 633, 635, 637, 637, 639, 638, 640, 641, 641, 643, 644, 645, 646, 647, 648, 649, 651, 650, 652, 653, 654, 655, 657, 656, 658, 658, 659, 661, 663, 663, 664, 665, 666, 667, 668, 669, 669, 671, 672, 673, 674, 676, 677, 678, 679, 680, 681, 682, 682, 683, 685, 685, 687, 688, 688, 688, 690, 691, 692, 693, 694, 695, 697, 696, 698, 699, 701, 701, 703, 703, 704, 705, 706, 708, 708, 709, 710, 712, 711, 714, 713, 716, 716, 717, 719, 720, 721, 722, 722, 724, 724, 725, 726, 727, 728, 729, 730, 732, 732, 734, 734, 735, 735, 738, 739, 739, 741, 741, 742, 744, 744, 745, 746, 747, 748, 750, 750, 751, 753, 752, 754, 756, 756, 756, 758, 759, 760, 760, 762, 763, 764, 765, 766, 766, 768, 769, 770, 772, 771, 773,

774, 775, 777, 778, 779, 779, 780, 782, 782, 784, 784, 785, 787, 786, 789, 789, 791, 791, 793, 793, 794, 796, 796, 796, 797, 800, 800, 801, 803, 802, 805, 804, 806, 807, 808, 810, 810, 811, 812, 814, 814, 815, 816, 818, 818, 819, 820, 820, 822, 823, 823, 826, 826, 828, 828, 829, 829, 831, 832, 833, 833, 835, 836, 838, 837, 839, 841, 841, 842, 844, 844, 845, 846, 847, 848, 849, 851, 851, 851, 854, 855, 855, 856, 858, 858, 859, 859, 862, 863, 864, 863, 865, 867, 868, 868, 869, 870, 871, 873, 873, 874, 876, 876, 878, 878, 879, 880, 881, 882, 883, 884, 884, 887, 887, 889, 889, 889, 892, 892, 893, 894, 896, 896, 898, 897, 899, 901, 902, 903, 904, 904, 905, 907, 907, 908, 909, 911, 911, 913, 912, 915, 915, 917, 918, 918, 918, 921, 922, 922, 923, 924, 924, 927, 927, 928, 929, 930, 931, 931, 932, 935, 936, 936, 936, 938, 940, 940, 941, 942, 943, 943, 945, 946, 947, 949, 950, 950, 951, 952, 953, 954, 955, 957, 958, 959, 959, 960, 962, 963, 963, 963, 965, 966, 968, 968, 969, 970, 972, 973, 972, 975, 975, 975, 977, 978, 979, 981, 980, 981, 981, 984, 986, 986, 987, 989, 989, 990, 990, 992, 993, 995, 996, 996, 998, 998]

Зависимость порядка матрицы от ее ранга представлена на (рис.2.4 приложения Г).

Также были рассмотрены следующие порождающие многочлены:

P(x)=$x^7 + x^4 + x^3 + x^2 + 1$, P(x)=$x^{13} + x^8 + x^5 + x^3 + 1$.

P(x)=$x^{24} + x^{11} + x^5 + x^2 + 1$, P(x)=$x^{25} + x^{13} + x^{12} + x^7 + 1$.

P(x)=$x^{41} + x^{32} + x^{31} + x^{27} + 1$, P(x)=$x^{39} + x^{35} + x^{23} + x^{16} + 1$.

Введем следующие обозначения:

- L - степень порождающего многочлена регистра сдвига;
- r – ранг матрицы;
- k – некий коэффициент, связанный с количеством операций;
- m – индекс суммирования;
- c_m – коэффициенты, которые могут быть равны либо 1, либо 0, в зависимости от элементов матрицы.

Основные элементы:

- **Ранг матрицы** — это максимальное количество линейно независимых строк или столбцов в матрице.
- **Порядок** — количество операций, которые необходимо выполнить в процессе формирования матрицы (это может быть связано с числом независимых строк и столбцов).
- **Размерность матрицы** — число строк и столбцов. Мы будем рассматривать матрицы размером $2^L \times 2^L$, где L — параметр, определяющий размер матрицы.

Общий вид:

Для самосжимающего генератора зависимость порядка от ранга можно представить в следующем виде:

$$r = 2^{L-m}k + 2^{L-m-1}$$

Таблица 2 зависимости

Ранг матрицы (r)	Порядок l
1	$2^{L-1}k$
2	$2^{L-1}k + 2^{L-2}$
4	$2^{L-2}k + 2^{L-3}$
…	…
2^m	$2^{L-m}k + 2^{L-m-1}$
… (*)	…
2^{L-1} – L + 2	2k+1 (**)

- Здесь k∈ N∪{0};
- (*) – Начиная с определенного момента, ранг равен степени двойки минус некоторое число. Экспериментальные данные показывают, что значение ранга не превосходит указанной в таблице величины;
- (**) - Порядок: 2k+1 – это особый случай, связанный с переходом генератора к последовательностям с короткими циклами. Порядок здесь всегда нечётный, что указывает на специфическую форму рекуррентной зависимости.

Возьмем полином 18 степени P(x)= $x^{18} + x^{16} + x^{11} + +x^4 + 1$ и продемонстрируем зависимость.

Для m=8, 2^8=256, в выходной последовательности рангов видно, что на 512 позиции точка "выпадает" и принимает значение 256.

Подставим в формулу: $2^{18-8}k + 2^{18-8-1} = 1024k + 512$, то есть ранг 256 будет у $1024k + 512$ (в данном случае k=0), следовательно, получаем 512, что и требовалось показать. В приведенной последовательности это можно наблюдать только на 512 позиции, так как размерность матриц не превышает 2^{20}=1024.

Рассмотрим m=9, 2^9 =512, в выходной последовательности рангов видно, что на 768 позиции точка "выпадает" и принимает значение 512.

Подставим в формулу: $2^{18-9}k + 2^{18-9-1} = 512k + 256$ (в данном случае k=1), следовательно, получаем, что ранг будет равняться 512 на 768 позиции, что и требовалось показать.

На рангах меньших степеней демонстрировать данную закономерность нецелесообразно, так как количество выпавших точек увеличивается в разы. На примитивных многочленах больших степеней эти закономерности не срабатывают, так как не хватает ресурсов для вычисления периодов. Следовательно, эти закономерности видны только в том случае, если был достигнут период.

Наблюдается общая тенденция: ранг матрицы увеличивается с ростом размерности, но остается ограниченным некоторым максимумом, зависящим от степени порождающего многочлена.

Линейная зависимость: Для малых размерностей наблюдается линейная зависимость ранга матрицы от размерности (диагональная линия на графике). Это характерно для начальной стадии генерации последовательности, когда матрица еще не достигает полного насыщения.

Насыщение ранга: С увеличением размерности матрицы ранг достигает плато — на графике это горизонтальные линии. Достижение плато указывает на то, что ранг ограничен степенью порождающего полинома.

Редкие значения: Наблюдаются точки ниже диагонали и горизонтальных плато, что соответствует свойствам самосжимающего генератора: некоторые биты или комбинации битов игнорируются, что приводит к уменьшению ранга на отдельных шагах.

Вывод:

Ранг матрицы для самосжимающего генератора, заданного полиномами различных степеней, растет с увеличением размерности матрицы, но ограничивается значением, которое зависит от **степени порождающего полинома**. Чем выше степень полинома, тем выше ранг матрицы на насыщении.

ЗАКЛЮЧЕНИЕ

В ходе дипломной работы была проведена всесторонняя теоретическая и практическая оценка работы криптографических генераторов псевдослучайных чисел, основанных на применении регистров сдвига с линейной обратной связью (LFSR). Особое внимание было уделено анализу двух разновидностей генераторов – прореживающего и самосжимающего – с целью выявления их характеристик, преимуществ, недостатков, а также их потенциальной пригодности для использования в системах защиты информации.

Работа охватывает широкий спектр задач, начиная от изучения математической природы и структуры регистров сдвига, построения соответствующих генераторов и заканчивая глубоким экспериментальным анализом получаемых выходных последовательностей. На основании разработанных алгоритмов были сформированы матрицы над конечным полем полем GF(2), ранги которых использовались в качестве нелинейных признаков для анализа псевдослучайности. Проведенные вычисления показали наличие устойчивой зависимости между порядком матрицы и ее рангом, что позволило более глубоко оценить качество выходных данных генераторов.

Для оценки статистических свойств сгенерированных последовательностей были использованы такие проверенные методы, как критерий Колмогорова-Смирнова и критерий Хи-квадрат. Оба теста подтвердили, что последовательности, генерируемые прореживающим и самосжимающим генераторами, демонстрируют признаки равномерного распределения и высокой степени случайности, что делает их потенциально пригодными для применения в криптографических целях.

Дополнительно, теоретически и экспериментально было подтверждена точность формул вычисления периода выходных последовательностей, что является важным критерием в оценке стойкости генераторов к атакам. Использование различных пар примитивных полиномов в качестве базовых элементов генераторов позволило продемонстрировать универсальность разработанных алгоритмов и их устойчивость к варьированию входных параметров.

Таким образом, были исследованы основные теоретические принципы работы генераторов на основе LFSR, проведён анализ их структуры и выходных данных, разработаны и реализованы программные модули, подтверждена псевдослучайность получаемых

последовательностей, а также установлены важные зависимости между параметрами, характеризующими выходные данные.

Результаты данной дипломной работы могут найти применение в таких областях, как проектирование защищённых каналов связи, генерация ключей шифрования, разработка безопасных программно-аппаратных решений и другие направления, требующие надёжных и качественных источников псевдослучайных данных.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

- 1.Харин Ю.С., Берник В.И., Матвеев Г.В., Агиевич С.В. Математические и компьютерные основы криптологии – 181 с.
- 2. Мальцев М.В., Палуха В.Ю., Харин Ю.С. О некоторых подходах к моделированию выходных последовательностей криптографических генераторов // Веб-программирование и интернет-технологии Web-Conf2012. Материалы 2-й Международной научно-практической конференции. 5 - 7 июня 2012 г., Минск. С. 133-134.
- 3. Поточные шифры. Результаты зарубежной открытой криптологии // Москва, 1997 – 24 с.
- 4. Caballero-Gil P., Fuster-Sabater A., Pazo-Robles M.E. New Attack Strategy for the Shrinking Generator //Journal of Research and Practice in Information Technology, Vol. 41, No. 2, May 2009 – 9с.
- 5. Слеповичев И.И. Генераторы псевдослучайных чисел – 139 с.
- 6. Харин Ю.С., Палуха В.Ю. Информативные признаки для статистического распознавания криптографических генераторов – 127 с.
- 7. Сушко.С.А. Практическая криптология.
- 8.Coppersmith, D., Krawczyk, H., & Mansour, Y. (1993). *The Shrinking Generator*. CRYPTO 1993. DOI: 10.1007/3-540-48329-2_2 – 28 с.
- 9. Menezes, A., van Oorschot, P., & Vanstone, S. (1996). *Handbook of Applied Cryptography*. CRC Press. Глава 6: Linear Feedback Shift Registers – 195 с.
- 10. Hardy, G.H., & Wright, E.M. (2008). *An Introduction to the Theory of Numbers*. Oxford University Press. Глава 8: «Congruences to Composite Moduli» - 94 с.
- 11. Meier, W., & Staffelbach, O. (1994). *The Self-Shrinking Generator*. EUROCRYPT 1994. DOI: 10.1007/BFb0053436 – 208 с.
- 12. Golomb, S.W. (1967). *Shift Register Sequences*. Holden-Day. Глава 3: «Linear Recurring Sequences» - 74 с.

ПРИЛОЖЕНИЕ А

Реализация на Python монобит-теста для прореживающего генератора:

```
import numpy as np
from scipy.stats import norm
def lfsr(taps, state):
    while True:
        xor = 0
        for t in taps:
            xor ^= state[t]
        yield state.pop(0)
        state.append(xor)
def monobit_test(sequence):
    n = len(sequence)
    s = sum(1 if bit == 1 else -1 for bit in sequence)
    s_obs = abs(s) / (n ** 0.5)
    p_value = 2 * (1 - norm.cdf(s_obs))
    return s_obs, p_value
# Параметры генераторов
L1, poly1 = 7, [0, 1]
L2, poly2 = 11, [0, 2]
state1 = [1] * L1
state2 = [1] * L2
# Создание генераторов
gen1 = lfsr(poly1, state1)
gen2 = lfsr(poly2, state2)
# Генерация последовательностей
at = [next(gen1) for _ in range(2000000)]
st = [next(gen2) for _ in range(2000000)]
# Формирование выходной последовательности xt
xt = [a for a, s in zip(at, st) if s == 1]
s_obs, p_value = monobit_test(xt)
print("Статистика S_obs:", round(s_obs, 4))
print("p-value:", round(p_value, 4))
if p_value >= 0.01:
    print("Последовательность прошла монобит-тест (равномерна)")
else:
    print("Последовательность не прошла монобит-тест (неравномерна)")
```

Реализация на Python монобит-теста для самосжимающего генератора:

```
import numpy as np
from scipy.stats import norm
def lfsr(taps, state):
    while True:
        xor = 0
        for t in taps:
            xor ^= state[t]
        yield state.pop(0)
        state.append(xor)
def monobit_test(sequence):
    n = len(sequence)
    s = sum(1 if bit == 1 else -1 for bit in sequence)
    s_obs = abs(s) / (n ** 0.5)
    p_value = 2 * (1 - norm.cdf(s_obs))
    return s_obs, p_value
L = 11
poly = [0, 1, 8, 10]
state = [1] * L
# Генератор LFSR
gen = lfsr(poly, state)
# Генерация самосжимающей последовательности
sequence_length = 1000000
compressed_sequence = []
while len(compressed_sequence) < sequence_length:
    first_bit = next(gen)
    second_bit = next(gen)
    if first_bit == 1:
        compressed_sequence.append(second_bit)
# Монобит-тест
s_obs, p_value = monobit_test(compressed_sequence)
print("Статистика S_obs:", round(s_obs, 4))
print("p-value:", round(p_value, 4))
if p_value >= 0.01:
    print("Последовательность прошла монобит-тест (равномерна)")
else:
    print("Последовательность не прошла монобит-тест (неравномерна)")
```

ПРИЛОЖЕНИЕ Б

Реализация на Python генерации последовательности прореживающего генератора и формирование матриц:

```
def lfsr(taps, state):
    while True:
        xor = 0
        for t in taps:
            xor ^= state[t]
        yield state.pop(0)
        state.append(xor)
L1, poly1 = 7, [0, 1]
L2, poly2 = 11, [0, 2]
state1 = [1] * L1
state2 = [1] * L2
gen1 = lfsr(poly1, state1)
gen2 = lfsr(poly2, state2)
# Генерация последовательностей
at = [next(gen1) for _ in range(2000000)]
st = [next(gen2) for _ in range(2000000)]
# Формирование выходной последовательности xt
xt = [a for a, s in zip(at, st) if s == 1]
# Вывод последовательностей
print("Элементарная последовательность at:", at)
print("Управляющая последовательность st:", st)
print("Выходная последовательность xt:", xt)


import numpy as np
def binary_matrix_rank(sequence, max_k=1500):
```

```
def gf2_rank(matrix):
    matrix = matrix.copy()
    rows, cols = matrix.shape
    rank = 0
    for col in range(cols):
        # Find pivot
        pivot = -1
        for row in range(rank, rows):
            if matrix[row, col] == 1:
                pivot = row
                break
        if pivot == -1:
            continue
        # Swap rows
        matrix[[rank, pivot]] = matrix[[pivot, rank]]
        # Eliminate below
        for row in range(rank + 1, rows):
            if matrix[row, col] == 1:
                matrix[row] ^= matrix[rank]
        rank += 1
    return rank
ranks = []
for k in range(1, max_k + 1):
    if len(sequence) < k**2:
        break
    matrix = np.array(sequence[:k**2]).reshape(k, k)
    matrix %= 2
    print (f"Матрица для k= {k}:\n{matrix}\n")
```

```
        rank = gf2_rank(matrix)
        ranks.append(rank)
    return ranks
sequence =  [..]
ranks = binary_matrix_rank(sequence)
print(ranks)
```

Реализация на Python генерации последовательности самосжимающего генератора и формирование матриц:

```
        def lfsr(taps, state):
            while True:
                xor = 0
                for t in taps:
                    xor ^= state[t]
                yield state.pop(0)
                state.append(xor)
        L = 11
        poly = [0, 1, 8, 10]
        state = [1] * L
        gen = lfsr(poly, state)
        sequence_length = 1000000
        compressed_sequence = []
        while len(compressed_sequence) < sequence_length:
            first_bit = next(gen)
            second_bit = next(gen)
            if first_bit == 1:
                compressed_sequence.append(second_bit)
        print("Самосжимающая последовательность (xt):", compressed_sequence)
```

```
import numpy as np
def binary_matrix_rank(sequence, max_k=1500):
    def gf2_rank(matrix):
        matrix = matrix.copy()
        rows, cols = matrix.shape
        rank = 0
        for col in range(cols):
            # Find pivot
            pivot = -1
            for row in range(rank, rows):
                if matrix[row, col] == 1:
                    pivot = row
                    break
            if pivot == -1:
                continue
            # Swap rows
            matrix[[rank, pivot]] = matrix[[pivot, rank]]
            # Eliminate below
            for row in range(rank + 1, rows):
                if matrix[row, col] == 1:
                    matrix[row] ^= matrix[rank]
            rank += 1
        return rank
    ranks = []
    for k in range(1, max_k + 1):
        if len(sequence) < k**2:
            break
        matrix = np.array(sequence[:k**2]).reshape(k, k)
```

```
        matrix %= 2
        print (f"Матрица для k= {k}:\n{matrix}\n")
        rank = gf2_rank(matrix)
        ranks.append(rank)
  return ranks
sequence =  [..]
# Получаем последовательность рангов
ranks = binary_matrix_rank(sequence)
print(ranks)
```

ПРИЛОЖЕНИЕ В

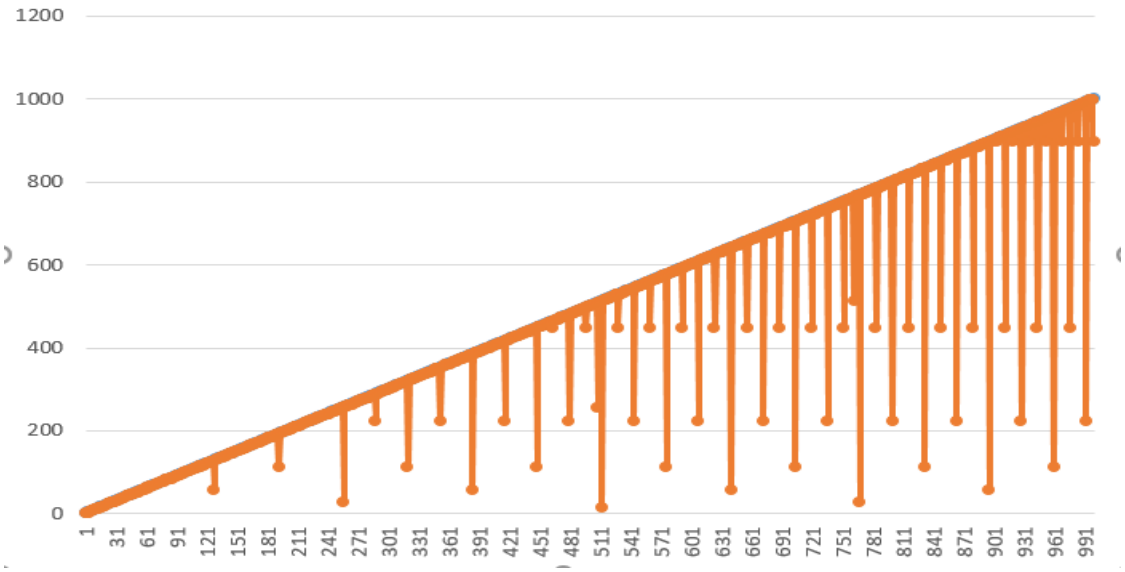
Реализация на Python вычисление периода для прореживающего генератора:

```
def lfsr(taps, state):
    while True:
        xor = sum(state[i] for i in taps) % 2
        yield state.pop(0)
        state.append(xor)
# Инициализация LFSR1 (P1(x) = x^3 + x + 1)
state1 = [1, 0, 0]
gen1 = lfsr([0, 1], state1.copy())
# Инициализация LFSR2 (P2(x) = x^2 + x + 1)
state2 = [1, 0]
gen2 = lfsr([0, 1], state2.copy())
output = []
for _ in range(100):
    s = next(gen1)
    if s == 1:
        output.append(next(gen2))
    else:
        next(gen2)
def find_period(seq):
    for period in range(1, len(seq)):
        if seq[:period] == seq[period:2*period]:
            return period
    return None
print("Период выходной последовательности:", find_period(output))
```

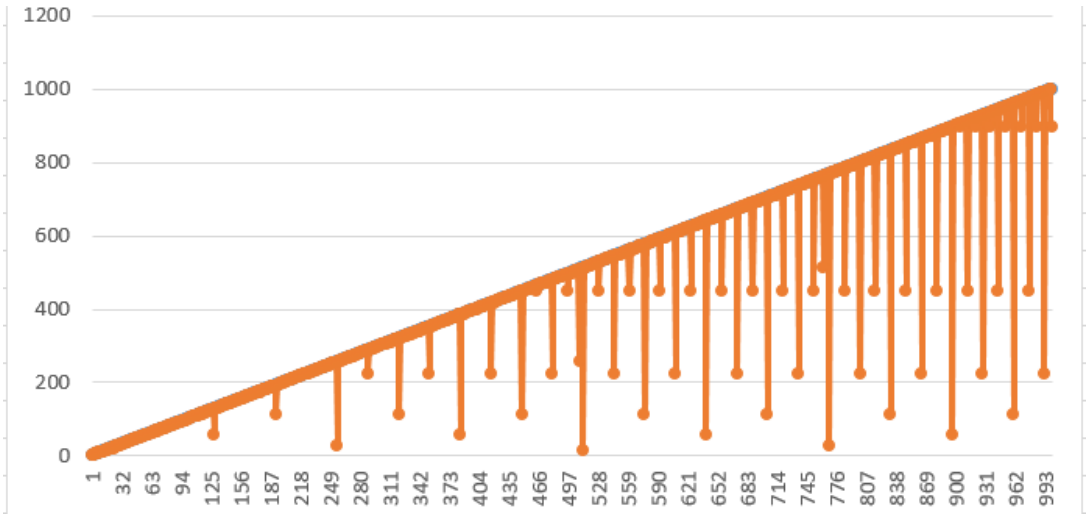
Реализация на Python вычисления периода для самосжимающего генератора:

```
class LFSR:
    def __init__(self, taps, state):
        self.taps = taps
        self.state = state.copy()
        self.length = len(state)
    def next_bit(self):
        feedback = 0
        for tap in self.taps:
            feedback ^= self.state[tap]
        self.state = [feedback] + self.state[:-1]
        return self.state[-1] if self.length > 1 else feedback
def self_shrinking_generator(lfsr, max_bits=10000):
    output = []
    initial_state = lfsr.state.copy()
    generated = 0
    while len(output) < max_bits:
        bit1 = lfsr.next_bit()
        bit2 = lfsr.next_bit()
        if bit1 == 1:
            output.append(bit2)
        generated += 2
        if generated > 2 * max_bits:
            break
    lfsr.state = initial_state
    return output
def find_period(sequence):
    if not sequence:
        return 0
    max_period = min(len(sequence) // 2, 10000)
    for period in range(1, max_period + 1):
        if all(sequence[i] == sequence[i % period] for i in range(len(sequence))):
            return period
    return len(sequence)
```

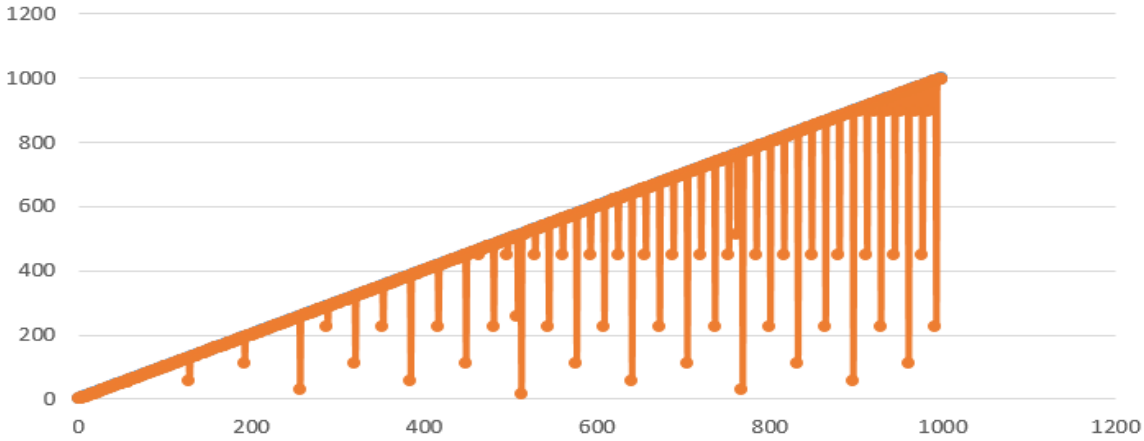
ПРИЛОЖЕНИЕ Г



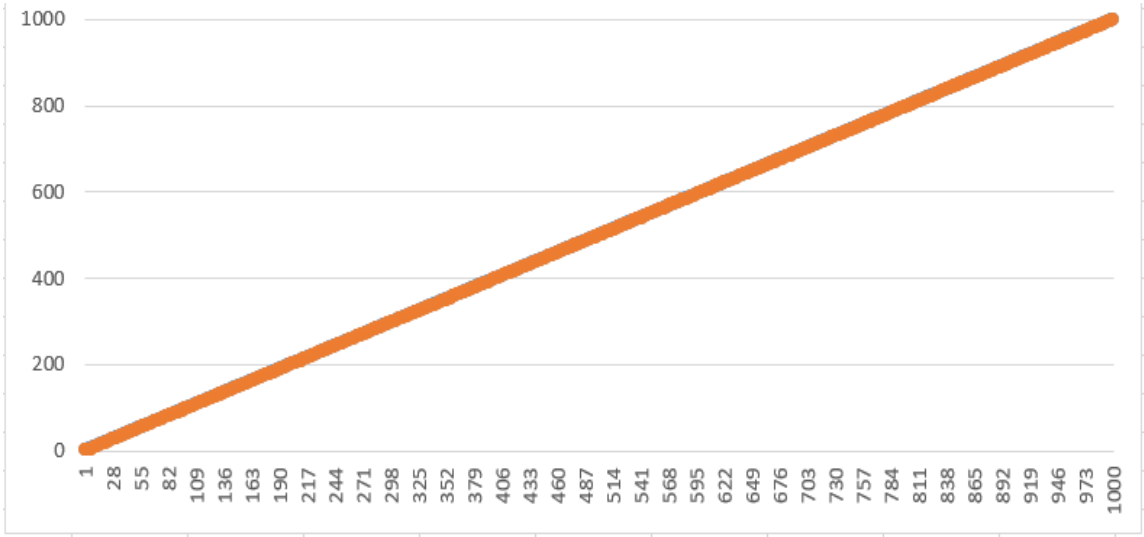
(рис. 1.1)



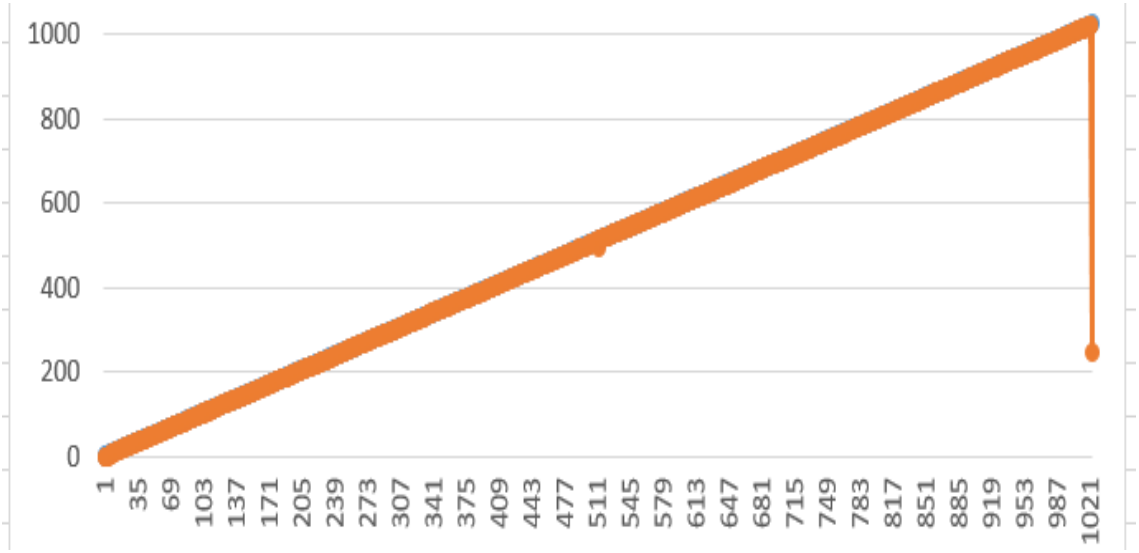
(рис. 1.2)



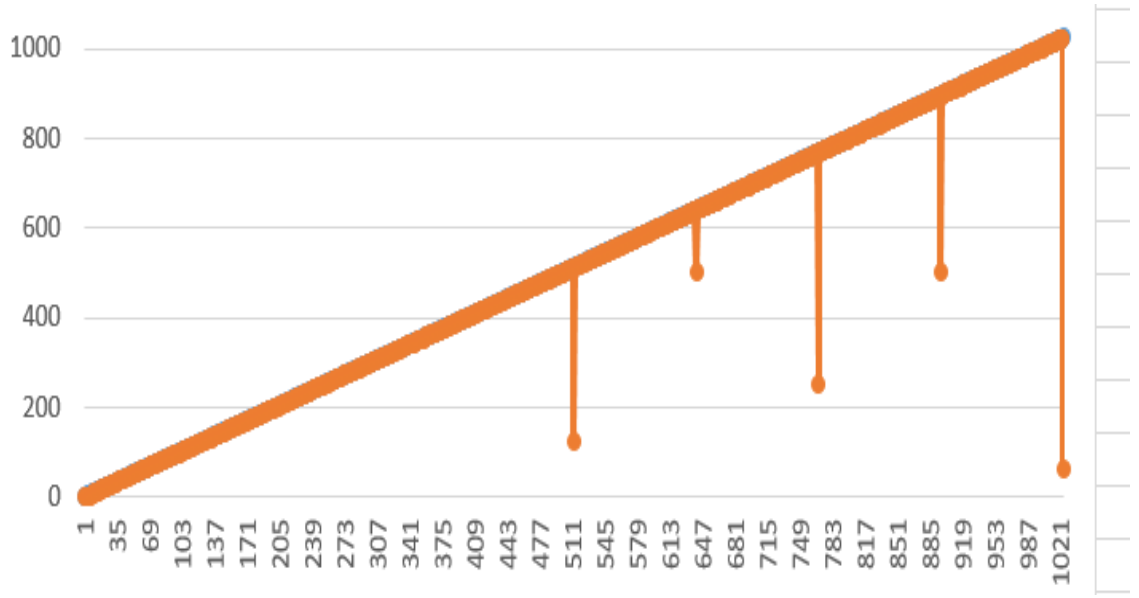
(рис. 1.3)



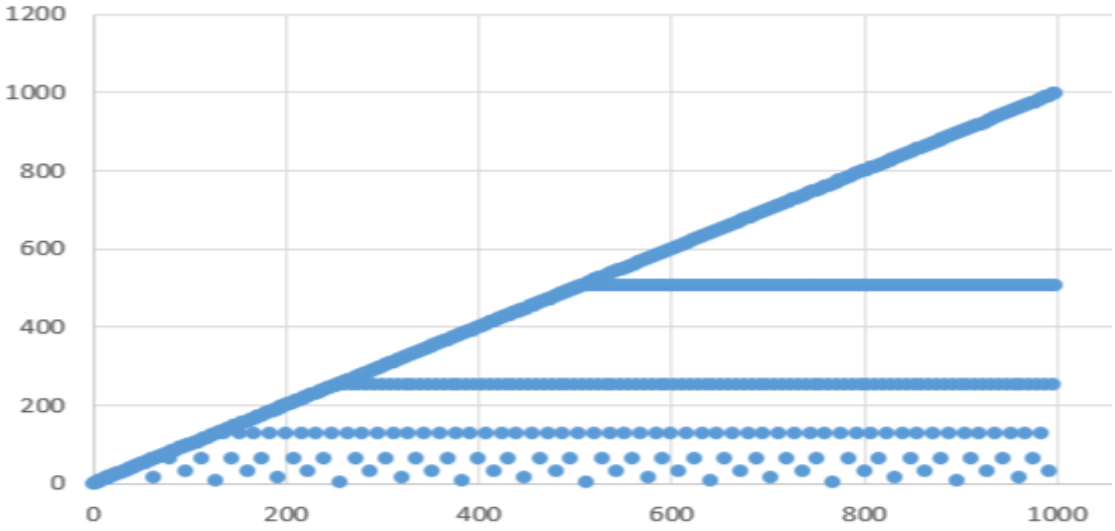
(рис. 1.4)



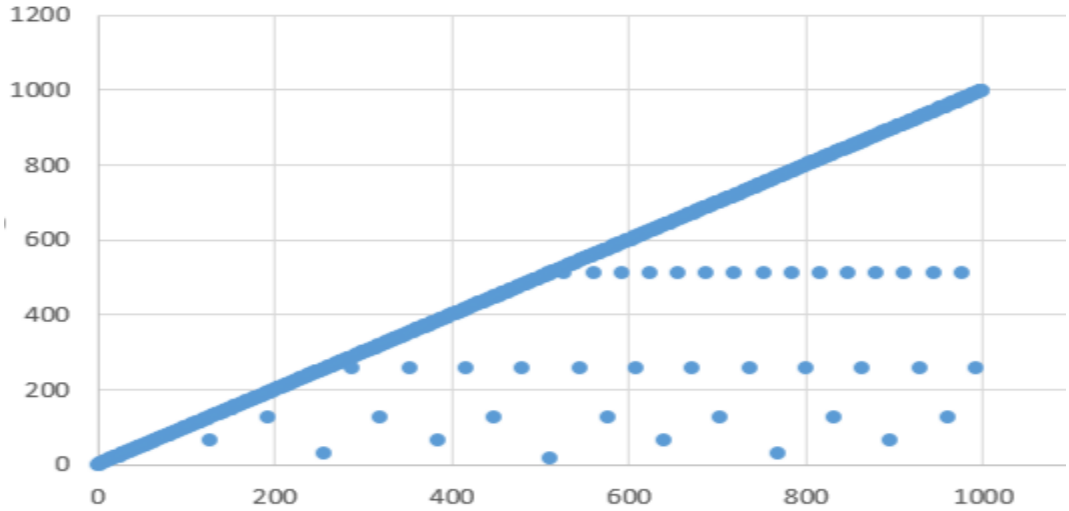
(рис. 1.5)



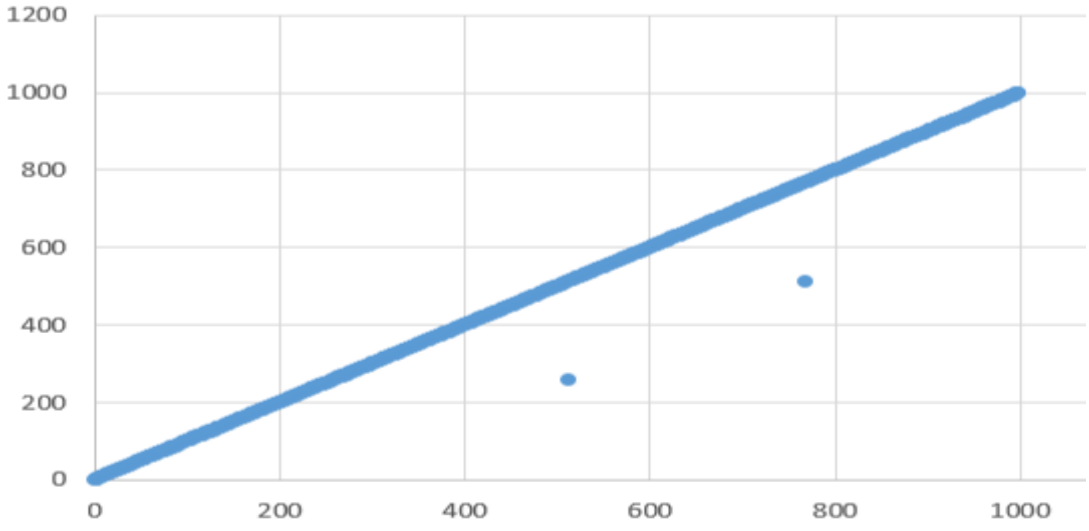
(рис. 1.6)



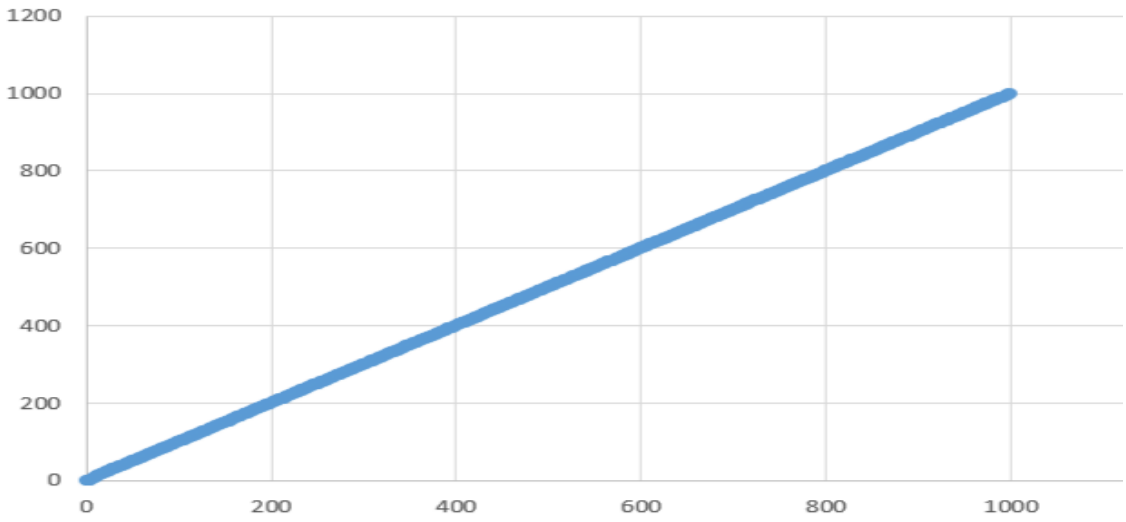
(рис. 2.1)



(рис. 2.2)



(рис. 2.3)



(рис. 2.4)

ПРИЛОЖЕНИЕ Д

k / n	3	5			7				
2	1								
3	1								
4	1								
5	2	1	2	3					
6	1	1	4	5					
7	1	2	3	4	1	2	3	4	5
8		1	2	7	2	4	5	6	7
9	4	3	5	6	2	3	6	7	8
10	3	2	3	8	1	2	5	6	7
11	2	1	8	10	1	2	5	7	9
12		1	2	10	2	6	8	9	10
13		3	5	8	1	2	5	10	12
14		1	11	12	1	3	4	5	11
15	1	3	4	12	5	6	7	8	13
16		10	12	15	2	9	12	13	14
17	3	4	12	16	2	5	6	8	13
18	7	4	11	16	1	4	7	8	10
19		3	9	10	6	12	13	16	18
20	3	2	7	13	1	10	14	16	18
21	2	3	4	9	6	8	14	18	19
22	1	3	7	12	2	4	9	14	21
23	5	4	8	15	5	11	12	13	17
24		2	5	11	3	6	7	16	23
25	3	7	12	13	7	10	13	15	23
26		13	15	23	1	6	15	17	24
27		17	22	23	6	11	17	18	19
28	3	5	8	24	5	11	21	24	27
29	2	2	6	16	3	11	15	16	22
30		9	10	27	11	12	24	28	29
31	3	8	23	25	1	8	10	14	16
32		2	7	16	1	3	12	17	30
33	13	11	16	26	1	8	17	19	32
34		8	12	17	4	7	14	20	31
35	2	9	17	27	2	21	23	31	32
36	11	7	12	33	6	17	25	26	28
37		2	14	22	3	21	30	31	33
38		5	6	27	6	9	11	20	36
39	4	16	23	35	2	13	15	36	38
40		23	27	29	6	7	18	28	36
41	3	27	31	32	11	12	20	32	40
42		30	31	34	1	8	14	24	27
43		5	22	27	8	25	30	32	35
44		18	35	39	5	16	25	40	43
45		4	28	39	14	15	23	27	33
46		18	31	40	21	23	24	40	44
47	5	11	24	32	5	17	19	32	42
48		1	9	19	5	12	27	29	43
49	9	16	18	24	8	39	41	42	45
50		17	31	34	5	6	16	21	36
51		15	24	46	12	15	22	24	25

k / n	3	5			7				
52	3	17	18	22	1	2	16	25	50
53		20	41	50	4	18	29	37	51
54		29	49	53	9	10	23	24	34
55	24	19	38	50	16	23	44	45	51
56		29	39	41	5	20	28	38	45
57	7	1	16	42	4	5	31	40	50
58	19	4	37	52	23	32	37	54	55
59		26	46	54	21	22	34	45	53
60	1	27	28	34	12	13	19	31	48
61		15	19	44	33	38	47	52	59
62		3	26	57	2	9	16	18	48
63	1	20	44	54	5	8	18	22	60
64		9	34	61	23	28	31	56	61
65	18	10	18	38	8	10	15	43	60
66		39	48	55	4	7	8	23	50
67		3	33	61	25	26	28	44	64
68	9	29	47	62	14	29	39	41	63
69		20	27	63	21	22	39	44	50
70		3	57	69	30	34	43	58	63
71	6	48	53	59	21	30	34	45	49
72		2	14	23	6	10	11	14	22
73	25	11	50	58	2	12	35	48	66
74		7	43	68	4	17	23	28	69
75		14	18	33	2	21	29	60	72
76		14	29	52	1	17	27	28	34
77		2	36	52	13	25	62	68	74
78		16	20	47	5	29	40	53	73
79	9	24	28	44	28	33	39	56	57
80		17	27	75	10	37	50	51	70
81	4	9	34	43	1	27	28	48	63
82		27	41	68	43	44	53	66	79
83		16	33	55	25	27	42	47	67
84	13	45	51	59	15	30	49	62	82
85		11	36	50	17	22	27	44	78
86		7	10	80	32	47	56	65	78
87	13	21	53	56	24	52	65	68	85
88		15	53	86	33	46	51	54	86
89	38	34	67	77	18	21	31	68	81
90		10	58	71	45	62	64	74	82
91		29	31	50	1	44	58	78	83
92		13	24	32	42	47	65	74	76
93	2	67	77	88	12	66	73	80	83
94	21	18	29	80	2	14	18	28	43
95	11	11	77	83	5	17	40	90	92
96		15	17	81	4	10	11	14	57
97	6	17	44	93	5	6	28	53	82
98	11	26	85	87	5	34	35	41	75
99		11	38	68	4	9	28	43	84
100	37	36	60	81	16	22	34	77	83
101		26	74	83	33	45	57	86	92