Міністерство освіти і науки України

Харківський національний університет імені В.Н.Каразіна

Факультет комп’ютерних наук

Кафедра безпеки інформаційних систем і технологій

Курсова робота

З дисципліни: «Спеціалізовані мови програмування та проектування електронних елементів і систем»

На тему: «Генерація криптографічних булевих функцій для нелінійних вузлів ускладнення симетричних шифрів із застосуванням методів імітації відпалу»

Виконав

Студент групи КБ-41

Кравченко Є. М.

Перевірив:

Професор

Олійников Р. В.

Харків – 2021 р.

**ЗМІСТ**

[**ВСТУП** 3](#_Toc70709660)

[**I.** **ДОСЛІДЖЕННЯ МЕТОДУ ІМІТАЦІЇ ВІДПАЛУ** 4](#_Toc70709661)

[**1.1.** **Покращення функції вартості** 4](#_Toc70709662)

[**II.** **ПРОГРАМНА РЕАЛІЗАЦІЯ** 9](#_Toc70709663)

[**ВИСНОВКИ** 10](#_Toc70709664)

[**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ** 11](#_Toc70709665)

# **ВСТУП**

Тема курсової роботи є досить актуальною у сучасному світі. Генерація булевих функцій для блочних шифрів є розповсюдженим методом.

# **ДОСЛІДЖЕННЯ МЕТОДУ ІМІТАЦІЇ ВІДПАЛУ**

В цьому розділі буде розглянуто можливі методи покращення імітації відпалу та експериментальні дослідження для деяких алгоритмів шифрування таких як ГОСТ 28147-89, MacGuffin, DES. Будуть проведені дослідження нових критеріїв обчислювального методу пошуку криптографічно стійких -боксів. На основі покращення вагових коефіцієнтів цінової функції пошуку запропоновано подальший розвиток методу імітації відпалу. В основу нових функцій запропоновані спектральні і кореляційні властивості недвійкових криптографічних функцій. Приведені оцінки швидкодії формування нелінійних вузлів замін методом імітації відпалу для деяких шифрів в порівнянні з методом випадкової генерації.

* 1. **Метод імітації відпалу**

Метод імітації відпалу – це алгоритм, заснований на локальному пошуку, багато в чому подібний більш складній формі альпінізму. Він заснований на техніці, яка використовується в металургії для усунення дефектів кристалічних структур в зразках металу.

При імітації відпалі деякий початковий сплав-кандидат, , зазвичай вибирається випадковим чином, вводиться в алгоритм SA разом із наступними параметрами:

Функція вартості.

Початкове значення для “температури”. Чим вище температура в поточній ітерації, тим більша ймовірність того, що алгоритм пошуку прийме погіршуючий хід, який призводить до прийняття кандидата з вищою вартістю, ніж поточний кандидат (тобто, зберігати зазначене рішення кандидата як "поточного кандидата"). Температура падає з часом, змушуючи алгоритм приймати менше неудосконалювальних ходів і, отже, перейти від розвідки до оптимізації. Ближче до кінця пошуку, вкрай рідко алгоритм приймає неудосконалювальний хід, і його поведінка дуже нагадує поведінку алгоритму сходження на гору.

Вибираючи значення , різні джерела стверджують, що його слід вибирати так, щоб певна частина погіршуючих ходів приймалася при температурі . Дуже мало інформація щодо того, якою має бути ця пропорція. Але зустрічаються варіанти такі як 0.6 або 0.8.

Значення  ­ "коефіцієнт охолодження", що визначає, наскільки швидко буде знижуватися температура після кожної ітерації алгоритму.

Ціле значення: MIL(MAX INNER LOOPS) визначає кількість локальних ходів алгоритму пошуку при кожному значенні температури.

Також повинен бути вказаний критерій зупинки. В даній реалізації використовувалися MAX OUTER LOOPS ­ значення, що вказує, скільки разів алгоритму слід було дозволити зменшувати температуру і продовжувати пошук до того, як він зупиниться.

## **Покращення функції вартості**

Одним з головних компонентів алгоритму імітації відпалу є цінова функція. Вона базується на обчисленні, на кожному кроці алгоритму формування - блоку, нелінійності, оцінюючи при цьому якість проведених покращень.

Розглянемо процедури формування цінових функцій, які використовуються для синтезу -блоків через спектральні характеристики булевих функцій, введемо відповідні функції вартості для синтезу -блоків через спектри недвійковий криптографічних функцій.

Нехай  задає -блок розмірності . Нехай для ,, – лінійна комбінація  виходів -блоку . Тоді  – значення перетворення Уолша-Адамара і значення автокореляції для кожної булевої функції .

Оскільки нелінійність булевої функції



то задача підвищення нелінійності може бути представлена як задача мінімізації абсолютного максимального значення коефіцієнтів Уолша-Адамара. Спочатку для синтезу -блоків використовували критерій нелінійності, тобто використовувалась наступна функція вартості:



Оскільки завдання пониження автокореляції представляє собою задачу мінімізації максимального значення автокореляційної функції, то  функція в подальших дослідженнях прийняла наступний вигляд:



Зазвичай в багатокритеріальних завданнях використовують такий підхід: обчислюється сума декількох окремих  функцій (по різним критеріям), помножених на вагові коефіцієнти. Тоді  функція у задачі пошуку -блоків з високою нелінійністю і низькою автокореляцією приймає вигляд:



Подальші покращення функції, були обґрунтовані наступним виразом.

Відомо, що рівність Парсеваля



обмежує  значенням рівним як мінімум . Ця межа досягається тоді, коли виконується рівність  для кожного . Коли значення деякого коефіцієнта  менше цієї ідеальної мажі, теорема Парсеваля стверджує, що інші значення коефіцієнтів  повинні бути вище цієї межі. Таким чином, спроба обмежити віддаленість абсолютних значень коефіцієнтів Уолша-Адамара від цієї межі може бути механізмом досягнення високої нелінійності. Спектри деяких функцій містять усі значення (по модулю), рівні цій ідеальній межі. Такі функції називають бент-функціями.

Окрім найвищої нелінійності, ці функції також мають нульову автокореляцію. Отже, функція вартості



є можливим підходом до оптимізації нелінійності та автокореляції. Дивлячись на незбалансованість бент-функцій представлена функція вартості  може бути покращена для знаходження збалансованих криптографічних функцій. В загальному вигляді функція (1.6) приймає вигляд



Параметри  та , називаються ваговими коефіцієнтами та забезпечують свободу експериментального дослідження та пошуку нових значень.

По аналогії з функціями вартості відносно спектра Уолша-Адамара виду (1.7), функція витрат відносно спектра автокореляційної функції матиме наступний вигляд:



Зазвичай, функції вартості використовуються для оптимізації окремих булевих функцій. Для всього ж вузла замін  функції, що базуються на спектрі Уолша-Адамара, можна узагальнити наступним чином:



і аналогічно для  функцій, основаних на автокореляційному спектрі:

.

Для оптимізації по критеріям нелінійності та автокореляції використовують наступну функцію вартості:



Спочатку в дослідженнях використовувалися функції вартості виду (1.8), (1.9), із заміною спектральних коефіцієнтів Уолша-Адамара і коефіцієнтів автокореляційних спектрів булевих функцій на коефіцієнти відповідних спектрів недвійкових функцій.

Подальші кроки дослідження полягали в удосконаленні функцій вартості (критерію пошуку криптографічних функцій), які ґрунтуються на наступному положенні. Відомо, що при оптимізації криптографічної функції по нелінійності і автокореляції вона по своїм спектральним характеристикам (спектру кореляції з лінійними функціями і автокореляційного спектру) прагне до спектральних характеристик бент-функцій, що і було використано в попередніх роботах [10; 15] при розробці функцій виду (1.6) - (1.11). Очевидним недоліком такого підходу є використання одного (фіксованого) значення статичного коефіцієнта, до якого прагнуть усі спектральні значення криптографічної функції, що оптимізується. При цьому значення спектральних коефіцієнтів ідеальної функції (або бент-функції) складаються з двох можливих значень для булевих функцій, і з трьох значень недвійкових функцій. При введенні ж додаткового обмеження на збалансованість, кількість можливих значень спектральних коефіцієнтів ще більше зростає. При розробці нових функцій вартості пропонується в (1.7) - (1.11) замінити статичний ваговий коефіцієнт  на так звані динамічні вагові коефіцієнти, тобто вагові коефіцієнти, які беруть різні значення для різних вхідних індексів спектра. У даній роботі в якості значень динамічних вагових коефіцієнтів використовуються спектральні значення бент-функцій. Пропоновані функції вартості мають вигляд:







де  – спектральні значення нелінійності і автокореляції випадкової недвійкової бент-функції .

Таким чином, в основі пропонованого обчислювального методу синтезу регулярних нелінійних вузлів замін симетричних криптоалгоритмів лежить застосування математичного апарату недвійковий криптографічних функцій [3], методів кореляційного і спектрального аналізу, а також запропонованих в даній роботі вдосконалених цінових функцій (1.12) - (1.14), з використанням динамічних вагових коефіцієнтів . Вдосконалений таким чином метод імітації відпалу дозволяє, як показано нижче, реалізувати обчислювальний пошук регулярних вузлів замін c необхідними показниками

нелінійності і автокореляції.

## **Експериментальні дослідження формування нелінійних вузлів замін**

Для підтвердження достовірності та обгрунтованості отриманих теоретичних результатів в даній роботі проведені експериментальні дослідження ефективності пропонованого обчислювального методу синтезу регулярних нелінійних вузлів замін. Перша частина досліджень була проведена з використанням спектрів недвійковий функцій зі статичними ваговими коефіцієнтами у функціях вартості (1.9) - (1.11) методу імітації відпалу, друга частина досліджень – з використанням динамічних коефіцієнтів у функціях цінності (1.12) - (1.14).

Шляхом експериментальних досліджень були обрані наступні параметри для обчислення:

 = 0.95 – параметр геометричного охолодження;

 = 500 – число кроків, виконуваних у внутрішньому циклі;

 = 50 – максимальне число внутрішніх циклів пошуку

 = 50 – максимальне число послідовних циклів, без внесення покращень;

В ході експериментів з різними функціями вартості виду (1.9) – (1.11) використовувалися різні значення статичного коефіцієнта  і фіксоване значення параметра  = 3.

1. **ПРОГРАМНА РЕАЛІЗАЦІЯ**

Для проведення досліджень описаних у минулому розділі було розроблено і реалізовано програмне рішенні мовою програмування C++. Програмна реалізація представляє собою консольний застосунок, який при запуску приймає текстовий файл з параметрами генерації, які більш детально будуть описані далі, а на виході файл із згенерованим -блоком, його параметрами, та часом витраченим на генерацію.

## 2.1. Аргументація вибору мови програмування

C ++ – надзвичайно потужний мова, що містить засоби створення ефективних програм практично будь-якого призначення, від низькорівневих програм та драйверів до складних програмних комплексів самого різного призначення. Ця мова програмування має багато переваг, але найбільш цінна для генерації вузлів замін – це швидкодія. Завдяки цьому будь-який алгоритм написаний мовою програмування C++ буде працювати швидше ніж на будь-якій іншій високорівневій мові програмування. Але за це розробники змушені жертвувати легкістю і швидкістю розробки.

## **2.2. Функціональність та призначення**

Програма розроблена для генерації нелінійних вузлів замін методом імітації відпалу, крім цього для порівняння дієвості алгоритму булі реалізовані основні популярні та дієві методи евристичного пошуку нелінійних вузлів замін, такі як Hill Climbing, genetic algorithm, genetic and tree algorithm.

Також в програмі реалізовані методи обчислення основних показників стійкості, їх опис представлено нижче.

Нелінійність (NL) – мінімальна відстань Хемінга між вихідною послідовністю *S* і всіма вихідними послідовностями афінних функцій та значення функції автокореляції, що забезпечує витік інформаційного потоку із входу на вихід функції. Внутрішні переходи між поданим на входи відкритим текстом і одержаним на виході закритим текстом в нелінійних вузлах замін описуються сукупністю компонентних булевих функцій.

Значення функціїавтокореляції(АС) *–* максимальне за модулем значення кореляції до всіх вхідних векторів.

Алгебраїчний імунітет – визначається як мінімальна ступінь багаточлена з ідеалу .

Збалансованість – рівність числа нулів і одиниць у вихідній послідовності.

Бажані параметри генерованого вузла замін задаються у файлі config.txt. Також у ньому задаються параметри генерації для усіх реалізованих алгоритмів. Приклад файлу налаштувань представлено на рисунку 2.1.

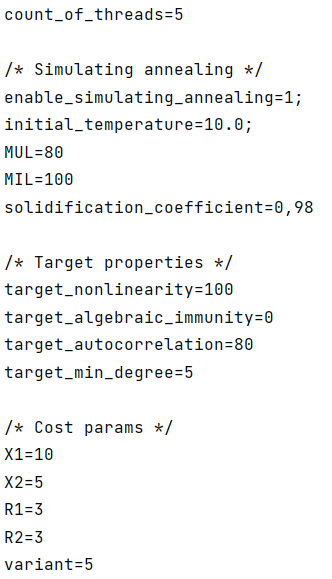


Рис. 2.1. – Файл налаштувань програмного продукту

Перш за все в файлі представлені основні налаштування системи.

count\_of\_thread – кількість потоків системи які одночасно будуть задіяні в обчисленнях та пошуку необхідного вузла замін. Якщо цього значення не буде, програма буде працювати у однопоточному режимі. Це буде набагато менш продуктивно ніж багатопоточний варіант. Для більшої швидкості роботи і використання всіх можливостей використованого комп’ютера, рекомендовано встановлювати значення рівне кількості ядер в системі або на один менше.

Далі йде блок в якому описані параметри, які повинен мати згенерований -блок. Ці параметри будуть узяті в якості критеріїв зупинки в алгоритмі імітації відпалу.

Наступний блок відповідає за параметри функції цінності . Вони необхідні для точного настроювання та тестування програми. В розділі з тестуванням нових цінових функцій ці параметри було змінено, для знаходження більш оптимальних значень.

Змінна variant відповідає яку конкретно функцію витрат буде використано, якщо ж вона буде відсутня використовується функція вартості за замовчуванням.

Взагалі, у кожного параметра представленого на рисунку 2.1 є значення за замовчуванням, вони встановлені з огляду на проведені тестування і відповідають оптимальним значенням. Це було зроблено для зручності використання. Якщо користувач запустить програму без файлу конфігурації він отримає попередження про те, що встановлено параметри за замовчуванням, і що, можливо, слід створити такий файл.

# **ВИСНОВКИ**

Отже в курсовій роботі було розглянуто різні види функцій цінності, проведений їх порівняльний аналіз. Було визначено що внесені покращення дають виграш у часі, знаходження вузлів замін методом імітації відпалу.

Також було реалізовано алгоритм генерації -блоків методом імітації відпалу. Програма вдалася дуже зручна для використання. Завдяки обдуманій архітектурі має широкі можливості для покращення, наприклад, можна швидко змінювати функції витрат, а також дописувати нові. Також перевагою розробленого програмного продукту є його зрозумілий інтерфейс та кінцеви результат.

# **ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Шнайер, Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. / Б. Шнайер. – Пер. с англ.: М.: Издательство ТРИУМФ, 2002. – 816 с.
2. Головашич, С.А. Анализ эффективности проектирования алгоритмов-участников конкурса БСШ Украины [Електронний ресурс] / С.А. Головашич // Х.: ООО КРИПТОМАШ, 2009. – С. 70. – Режим доступу до журн.: http://www.cryptomach.com/upload/ru/files/bc\_- design\_effectiveness.pdf.
3. Millan, W. Evolutionary Heuristics for Finding Cryptographically Strong S-Boxes / W.Millan, L. Burnett, G. Carter, A. Clark, E. Dawson // Information and communication security. – Springer, Heidelberg, 1999. Vol. 1726. – P. 263-274.
4. Kavut, S. Improved Cost Function in the Design of Boolean Functions Satisfying Multiple Criteria / S. Kavut, M.D.Yucel // Progress in ¨ Cryptology, INDOCRYPT 2003, 4th International Conference. – Springer, 2003. – Vol. 2904. – P. 121-134.
5. Dawson, E. Designing boolean functions for cryptographic applications / E. Dawson, W. Millan, L. Simpson // Proceedings of General Algebra Conference (AAA58). Verlag Johannes Heyn, 1999. – P. 1-22.