УДК 621.391.7

О. І. Цимбал

АНАЛІЗ МОЖЛИВОСТІ ВИКОРИСТАННЯ ГЕНЕТИЧНОГО АЛГОРИТМУ В КРИПТОГРАФІЧНИХ ШИФРАХ ДЛЯ ПІДВИЩЕННЯ ЇХ КРИПТОСТІЙКОСТІ

Вінницький національний технічний університет

Анотація

В роботі розглянуто та детально проаналізовано можливість використання математичного апарату генетичного алгоритму в криптографічних шифрах для підвищення їх криптостійкості. Описано основні властивості генетичного алгоритму та обґрунтовано його стійкість.

Ключові слова: генетичний алгоритм, криптографія, криптостійкість.

Abstract

The paper considers and analyzes the possibility of using the mathematical apparatus of the genetic algorithm in cryptographic ciphers to increase their cryptographic stability. The basic properties of the genetic algorithm are described and its stability is substantiated.

Keywords: genetic algorithm, cryptography, cryptostability.

Вступ

Криптографія в основному означає зберігати таємну або приховану інформацію. Існує ряд функцій, пов'язаних з криптографією. Однією з них є конфіденційність, яка в основному означає, що особи, які застосовують криптографічні алгоритми, методи та схеми повинні бути впевнені, що ніхто не матиме доступ напряму до їх інформації, коли вона перебуває в мережі. В даній галузі це можливо зробити лише розшифрувавши шифротекст, іншими словами зламати шифр або ж мати певний секретний ключ для проведення процесу дешифрування.

Метою роботи є аналіз можливості використання генетичного алгоритму в криптографічних шифрах для підвищення їх криптостійкості.

Результати дослідження

Генетичні алгоритми (ГА) базуються на концепції "виживання найбільш придатних" і працюють над тим, щоб знайти оптимальне або майже оптимальне рішення для оптимізації завдань.

Генетичні алгоритми це клас алгоритмів оптимізації. ГА має на меті вирішення завдань шляхом моделювання спрощеної версії генетичних процесів. Є багато проблем, для яких підхід ГА є корисним.

В цій роботі досліджується можливість використання ГА в криптографічному алгоритмі RSA для підвищення його криптостійкості. Як традиційний криптоаналіз, так і ГА-методи реалізовані в програмному забезпеченні. Результати потім порівнюються, використовуючи показники пройденого часу та відсоток успішних дешифрувань. Встановлюється визначення кожного розглянутого шифру стосовно обґрунтованості підходів, що базуються на ГА.

Нападів на генетичний алгоритм, знайденого в літературі, всього дванадцять, сім було повторно впроваджено. З цих семи лише троє досягли будь-якого успіху. Успішними нападами були ті, що стосувалися шифрів перенесення та перестановок. Ці напади були додатково вивчені в спробі покращити їх успіх. На жаль, ця спроба була невдалою, як і намагання застосувати інші типи атак [61].

Основна ідея ГА полягає в тому, щоб моделювати процес природного відбору, де застосовуються генетичні оператори для покращення генерації. Поетапно генетичний алгоритм виглядає наступним чином:

1. Оператор відбору: призначення цього оператора полягає у виборі кращих батьків, щоб передавати кращі характеристики до наступного покоління. Переваги кожного окремого покоління в певному поколінні залежать від його придатності, яка може бути розрахована об'єктивною функцією або суб'єктивним судженням.

2. Схрещення: за допомогою оператора виділення вибираються два найкращі об’єкти з множини об’єктів, а також вибирається випадкова точка схрещення. Біти міняються місцями в рядах бітів вибраних об’єктів, враховуючи випадкову точку схрещення [62].

Наприклад:

Якщо S1 = 11111111 і S2 = 00000000, а точкою схрещення було випадково вибрано 5, то S1'= 11111000 і S2' = 00000111. Очевидно, що процес схрещення, вироблятиме кращі набори біт, порівняно з вихідними рядками.

Оператор схрещення також поділяється на одиночне та подвійне схрещення.

Одиночне схрещення включає в себе обмін бітами по випадково вибраній точці схрещення, як показано на рис. 1 нижче.

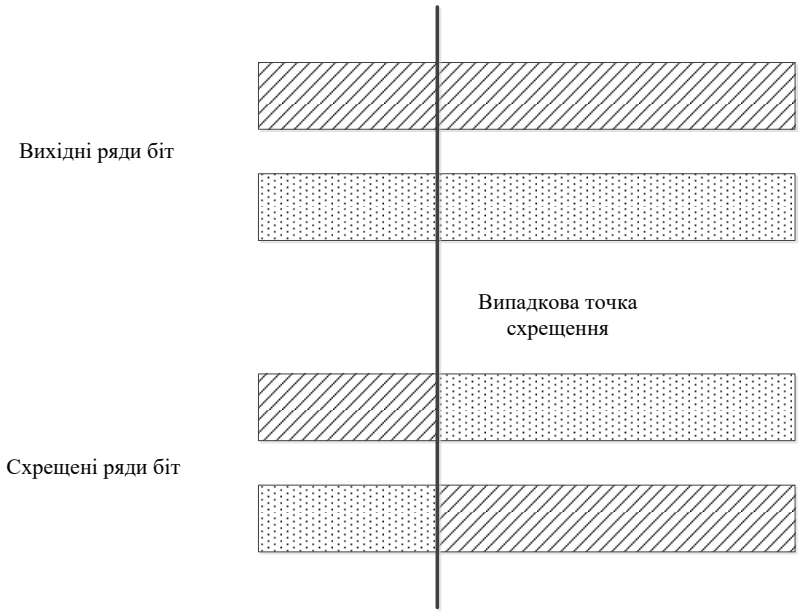

Рис. 1. Приклад одиночного схрещення

Подвійне схрещення включає в себе обмін бітами по випадково вибраних двох точках схрещення, як показано на рис. 2.

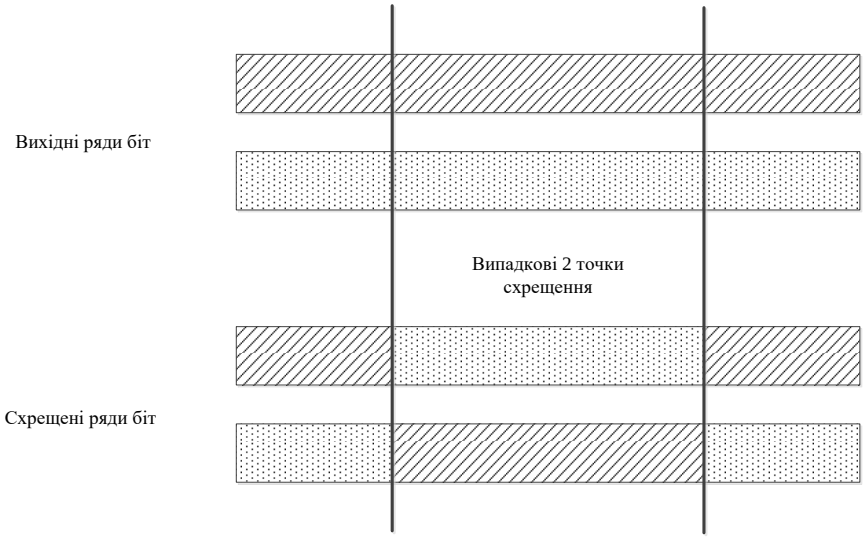

Рис. 2. Приклад подвійного схрещення

Мутація: деякі біти вихідних бітових рядків будуть перезаписані або змінені, щоб зберегти різноманітність.

Варто зазначити, що генетичному алгоритму притаманні такі чотири особливості:

- вони обробляють не значення параметрів самого завдання, а їх закодовану форму;
- здійснюють пошук рішення виходячи не з однієї точки, а з їх деякої популяції;
- використовують тільки цільову функцію, а не її похідні або іншу додаткову інформацію;
- застосовують імовірнісні, а не детерміновані правила вибору.

Перераховані чотири властивості, які можна сформулювати також як кодування параметрів, операції на популяціях, використання мінімуму інформації про завдання і рандомізація операцій

приводять у результаті до стійкості генетичних алгоритмів і до їх переваги над іншими широко вживаними технологіями.

Таким чином, використовуючи генетичні алгоритми при утворенні шифротексту чи генерації ключів, можна підвищити стійкість утворюваних послідовностей, що в свою чергу підвищить криптостійкість самого алгоритму шифрування інформації.

Висновки

У даній роботі було детально проаналізовано математичний апарат генетичного алгоритму. Описані його основні властивості та функції, такі як оператор відбору, мутації та схрещення. Обґрунтовано можливість його використання для підвищення стійкості криптографічних шифрів.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Гулак Г. М. Основи криптографічного захисту інформації / Г.М. Гулак, В.А. Мухачов, В.О. Хорошко, Ю.Є. Яремчук – Вінниця, 2011 – 199 с.

2. Азаров О. Д.. Комп'ютерна криптографія / Азаров О. Д., Хорошко В. О., Шєлєст М. Є., Мухачьов В. А., Яремчук Ю. Є. - НАУ, 2003 – 14 с.

3. Ю. Є. Яремчук. Сучасний захист інформації / Ю.Є. Яремчук, А. П. Бондарчук, С. Я. Довбня, Ю. І. Хлапонін – Вінниця, 2013.

4. Cryptography [Electronic Resource]. – Mode of access : URL : https://en.wikipedia.org/wiki/Cryptography - Назва з екрану.

5. Strong cryptography [Electronic Resource]. – Mode of access : URL : https://en.wikipedia.org/wiki/Strong_cryptography - Назва з екрану.

Цимбал Олександр Ігорович — магістр, Вінницький національний технічний університет, Вінниця, e-mail: ramonesz297@outlook.com.

Науковий керівник: ***Яремчук Юрій Євгенович*** — доктор технічних наук, професор, факультет менеджменту та інформаційної безпеки, Вінницький національний технічний університет, Вінниця.

Tsymbal Alexander Igorovich — master degree, Vinnitsa National Technical University, Vinnitsa, e-mail: ramonesz297@outlook.com.

Supervisor: ***Yaremchuk Yuriy E***. — Ph. D., professor, management and security of information Systems department; Vinnitsa.