DOI 10.18372/2310-5461.58.17650
УДК 004.8

О. О. Кубайчук, канд. фіз.-мат. наук, доцент,
Національний технічний університет України
"Київський політехнічний інститут
імені Ігоря Сікорського".
orcid.org/0000-0002-5135-3688
e-mail: o.kubaychuk@gmail.com

ОСОБЛИВОСТІ ЗАСТОСУВАННЯ АЛГОРИТМУ АСО ДО ДЕЯКИХ ЗАДАЧ КРИПТОАНАЛІЗУ

Вступ

Однією із загальновизнаних вимог до стійкості шифрів є правило, за яким число операцій, необхідних для розшифрування шля-хом перебору має перевищувати спроможності наявних комп'ютерів. Отже, задачу розшифру-вання можна розглядати як задачу комбінаторної оптимізації (КО). Теоретичні дослідження алгоритмів КО рідко дозволяють отримувати результати, які можуть бути застосовані на практиці. Тому основним інструментом аналізу їх ефективності є обчислювальний експеримент.

Задачі комбінаторної оптимізації виникають у багатьох областях застосування обчислювальних методів, зокрема, таких як, дослідження операцій, біоінформатика, маршрутизація, розподіл ресурсів і у криптоаналізі. Більшість практично важливих задач комбінаторної оптимізації належать до числа NP-важких, що, враховуючи можливі похибки у вхідних даних та можливість існування багатьох локальних екстремумів цільової функції, робить недоцільним використання точних алгоритмів розв'язання. Ці та інші аспекти, разом з прогресом обчислювальної техніки обумовили інтенсивний розвиток класу наближених методів, названих метаевристич-ними [1], які застосовуються до розв'язання задач КО. Метаевристичні методи в силу своєї структури і гнучкості дозволяють конструювати на базі єдиної обчислювальної схеми наближені алгоритми розв'язання доволі широкого класу задач за прийнятний час. Такі алгоритми називають метаевристиками. Метаевристика є стратегією, що ефективно управляє дослід-женням простору пошуку з метою отримання (суб)оптимального розв'язку. Дослідження прос-тору пошуку здійснюється, зокрема, на основі евристик – процедур (функцій), які не вимагають строгого теоретичного обгрунтування. Ефектив-ність тих чи інших евристик встановлюють емпі-ричним шляхом. Метаевристика не є проблемно-орієнтованою, але вона може використовувати знання з певної предметної області у формі евристик.

Значна частина існуючих метаевристичних алгоритмів відноситься до класу популяційних алгоритмів, - алгоритмів, у яких, на відміну від траекторних, на кожній ітерації опрацьовується не один, а одразу декілька варіантів розв'язку. Ця властивість популяційних алгоритмів є суттєвою причиною їх застосування до розв'я-зання задач криптоаналізу.

Постановка проблеми

На даний момент інтенсивно розвивається напрямок по застосуванню методів штучного інтелекту до розв'язання широкого кола задач, до яких відносяться і задачі криптоаналізу. Метаевристичні методи є методами штучного інтелекту. Тому дослідження особливостей зас-тосування метаевристичного пошукового алго-ритму АСО (Ant Colony Optimization) до деяких задач криптоаналізу є важливим і актуальним.

Однією з основних задач в області штучного інтелекту є задача алгоритмізації пошуку. Вибір тих чи інших пошукових алгоритмів значною мірою залежить від прийнятої моделі області задачі. До області задачі відносять множину станів та множину операторів над станами задачі. Наприклад, моделлю області задачі може бути деякий граф, вершини якого зображають стани, а ребра відповідають операторам.

Далі проводиться ретроспективний огляд задач криптоаналізу, сформульованих як задачі КО, метаевристичними методами.

Аналіз останніх досліджень та публікацій

Основні напрямки та результати сучасних досліджень методів комбінаторної оптимізації представлені Гуляницьким Л. Ф. та Мулесою О. Ю. в [1], причому, основну увагу приділено саме метаевристичним методам. Хо Єн Лі в [2] аналі-зує актуальні (на період до 2005 року) спроби застосування метаевристичного підходу до вирішення задач криптоаналізу класичних і блокових шифрів. У роботі [3], Тоймех і Арамахем пропонують оригінальну атаку на шифри перестановок, використовуючи генетич-ний алгоритм (GA). Приб-

лизно у той же період Сонг та ін. в [4] фокусуються на диференціаль-ному криптоаналізі шифру DES4, застосовуючи генетичний алгоритм. Герг у [5] доводить пере-ваги міметичного алгоритму перед генетичним у випадку атаки на SDES. Для атаки блокового шифру TEA, Вей Ху в [6] застосовує підхід, що поєднує квантові і еволюційні обчислення, а саме, використовує алгоритм QGA (Quantum Genetic Algorithm). Була показана ефективність QGA у випадку TEA4 і TEA5 у той час, як звичайний GA виявився неспроможним.

Атаку DES16 з використанням методу оптимізації роєм частинок (PSO) описано Елмонімом та ін. в [7]. Інший приклад застосування роєвого інтелекту – криптоаналіз класичних шифрів методом оптимізації колонією мурах (ACO) представлено Мехазні Т. та ін. у [8].

Боричка та ін. в [9] на прикладі класичних шифрів показують, що застосування еволюційних алгоритмів є доцільним для розв'язання задач криптоаналізу. Можливість атаки на DES4 за фрагментом шифротексту вивчають Дадліч та ін. у [10]. При цьому, для обчислення оптимального ключа використовуються роєві алгоритми оптимізації. Садехзадеш і Тагербахал, здійснюючи криптоаналіз шифрів перестановок, у [11] порівнюють ефективність використання детермінованого і стохастичного локального пошуку для при-скорення роботи генетичного алгоритму. Критичні зауваження щодо можливості застосування еволюційних алгоритмів для криптоаналізу блокового шифру SDES можна знайти у Тейтауда і Фонлапта [12].

Алгоритм пошуку, що імітує політ зозулі (Cuckoo Search) для обчислення ключа шифру Віженера, застосовано Ашоком та ін. [13]. Тахар у [14] дослідив можливість криптоаналізу шифрів SDES, DES4, DES за допомогою популяційного алгоритму оптимізації, що імітує полювання кажанів у повній темряві – алгоритму BAT. Дворак і Боричка у [15] описують диференціальний криптоаналіз блокового шифру FEAL4 (Four-rounded Fast Data Encipherment Algorithm) з використанням еволюційного алгоритму. Амік та ін. у [16] застосовують технологію роєвого інтелекту, що базується на поведінці колонії світляків (Binary Firefly Algorithm) до атаки на DES, а у [17] ці ж автори досліджують можли-вості метаевристики, що імітує поведінку домаш-ньої кішки (Binary Cat Swarm Optimization (BCSO)) до задач криптоаналізу.

До відкритого доступу потрапляє мало публікацій присвячених криптоаналізу актуальної криптосистеми з відкритим ключем RSA. Одним з перспективних напрямків у пошуках підходу до вирішення задачі факторизації є застосування методів комбінаторної оптимізації. Кандра, Ракшмаваті та ін. у [18] провели серію обчислювальних експериментів, які показали, що табуйований пошук можна застосовувати, але цей метод не є ефективним для розв'язання проблеми факторизації. Прикладом успішної атаки на ранцеву криптосистему Меркля-Хеллмана є застосування Грері та ін. у [19] алгоритму ACO. Згодом, той же колектив авторів описав можливість застосування ACO до криптоаналізу шифрів підстановки та блокових шифрів SDES і SAES у [20], [21] і [22]. Джейн та ін. у [23] порівнювали Cuckoo Search та GA, досліджуючи атаки на шифри підстановки. Алгоритм, що імітує стра-тегію зграї дельфінів під час полювання, застосовували Амік та ін. до шифру DES у [24]. Ракшмаваті та ін. [25] продовжили спроби вирішення проблеми факторизації, застосовуючи метаевристичний підхід. Цього разу в якості інструменту криптоаналізу було обрано генетич-ний алгоритм. Обчислювальні експерименти довели спроможність GA до атак на RSA, причо-му було показано, що підбір параметрів генетич-ного алгоритму має істотний вплив на швидкість отримання результату.

Сабончі і Акай у [26] в рамках диференціального криптоаналізу наводять можливості застосування еволюційного підходу та алгоритмів роєвого інтелекту до атак на класичні шифри. Вони ж у [27] до шифру Віженера, застосовують гібридизацію алгоритму оптимізації колонією бджіл, додаючи операцію біноміального кросоверу. В оглядовій статті [28] Сабончі та Акай підсумовують результати застосування різнома-нітних мета-евристик до криптоаналізу класич-них шифрів.

Серед останніх публікацій – роботи Грері та ін. [29], Дворак і Боричка [30], Джейн та ін. [31] які продовжили свої дослідження щодо застосування метаевристик у криптоаналізі.

Метою даного дослідження є аналіз особливостей застосування ACO для деяких задач криптоаналізу.

Виклад основного матеріалу

Важливим класом пошукових алгоритмів штучного інтелекту є клас багатоагентних алгоритмів, навіяних природою. Серед них чільне місце займають алгоритми ройового інтелекту (Swarm Intelligence), які експлуатують *емерджентні* властивості багатоагентних систем. До них належать алгоритми ACO – пошукові алгоритми, що імітують поведінку справжніх мурах. Першим з ACO-алгоритмів був алгоритм AS (Ant System) для розв'язання задачі комівоя-жера

(Traveling Salesman Problem, TSP) у 1992 році. Потім з'явилося декілька його модифікацій: кількісний (ant-quality), щільнісний (ant-density), циклічний (ant-cycle). Далі було запропоновано ряд потужніших алгоритмів ACO розв'язання TSP. Найвідомішими серед них є алгоритми на базі елітарної стратегії, алгоритми на основі ранжування та максимінний алгоритм мураши-них систем. Успіхи у розв'язанні задачі комівоя-жера стимулювали розробку алгоритмів ACO для вирішення інших, практично значимих задач дискретного програмування: транспортної задач, задачі про призначення, задачі календарного планування і задач криптоаналізу.

Алгоритми ACO показують задовільні результати для широкого класу задач, які зводяться до задачі пошуку на графах. Розглянемо задачу комбінаторної оптимізації (S, f, Σ), де S – множина кандидатів у розв'язки, Σ – множина обмежень, f – цільова (фітнес) функція, визначена для кожного $s \in S$ і, яка може залежати від часу (ітерації) t. Необхідно знайти глобальний мінімум.

Далі, нехай пошуковим простором для мурах є задана скінченна множина $V = \{v_1, v_2, \ldots v_{|V|}\}$. Формально, мураха є рандомізованим конструктивним алгоритмом, який будує розв'язок задачі оптимізації на графі $G = (V, E)$, де E – множина ребер графа. Мурахи будують допустимі розв'язки, рухаючись на G таким чином, щоб задовільнити обмеження Σ задачі.

З вершинами $v_i \in V$ і ребрами $e_{ij} \in E$ можна асоціювати феромонний слід τ (τ_i для вершин і τ_{ij} для ребер), тим самим кодуючи довготривалу пам'ять колонії про її попередній пошук. Також з вершинами і ребрами можна асоціювати значення евристики ω, що представляє собою деяку апріорну інформацію на вході задачі або нову інформацію, яка надходить в процесі розв'язання з джерел ніяк не пов'язаних з мурахами. Ймовірнісне правило вибору напрямку руху на графі використовує τ і ω.

Поведінку (*ant activity*) *k*-ї ($1 \le k \le M$) комахи, де M параметр алгоритму, можна описати так:

- Мураха досліджує граф $G = (V, E)$ та будує розв'язок (шлях) *s*, наприклад, мінімальної вартості;
- вона пам'ятає шлях, яким пройшла. Пам'ять, наприклад, можна використати для відкладення феромону під час повернення. Такий спосіб відкладення феромону називається *відтермінованим оновленням феромону* на відміну від *покрокового оновлення феромону*, який можна реалізувати на етапі приєднання нової компоненти розв'язку;
- для мурахи задається її початковий стан x_s^k та умова зупинки ε^k;
- Мураха, знаходячись у вершині i, шукає вершину j для переходу в допустимому околі O_i^k;
- Комаха рухається за ймовірнісним правилом, яке є функцією (*а*) локально доступного феромонного сліду та евристичної інформації, (*б*) особистої пам'яті про пройдений шлях, (*в*) обмежень задачі.

Важливо відзначити, що мурахи будують розв'язки незалежно одна від одної і кожна з них в змозі самостійно знайти розв'язок (можливо, далекий від оптимального) задачі. Як правило, якісні розв'язки виникають як результат колек-тивної взаємодії між мурахами через непряме спілкування, шляхом читання/запису у змінні, що зберігають значення слідів феромону. Тобто, відбу-вається розподілений процес навчання, в якому окремі агенти, мурахи, самі не адаптуються, але впливають на рішення інших учас-ників.

Метаевристики, крім основної процедури (у нашому випадку це діяльність (*ants activity*) мурах, описана вище), можуть містити декілька додаткових. Метаевристика ACO включає дві додаткові процедури: випаровування феромонного сліду (*pheromone trail evaporation*), демон (*daemon activity*, необов'язкова процедура). Випаровування феромонного сліду означає зниження впливу сліду з часом. З практичної точки зору, випаровування феромону допомагає уникнути передчасної збіжності до локального оптимуму. Так реалізований процес «забування» спонукає до переходу у нові області пошукового простору. Демон можна розглядати як деяку насильницьку інструкцію. Прикладом демона може бути активація якоїсь локальної оптимі-заційної процедури. Демоном для ACO метаев-ристики часто є відкладення більших значень феромону на пріоритетних напрямках дослід-жень. Таке оновлення феромону називається оф-лайн оновленням (*off-line pheromone updates*).

Метаевристики допускають абстрактний опис, оскільки не є проблемно-орієнтованими. Псевдокод метаевристики ACO має вигляд представлений на лістингу 1:

Лістинг 1:

```
АСО_МЕТАЕВРИСТИКА
ScheduleActivities
        AntsActivity()
        EvaporatePheromone()
        DaemonActions() {Опціонально}
    end ScheduleActivities
end
```

Відзначимо, що в конструкції *Schedule Activities* (план/розклад дій), порядок виконання та синхронізація процедур нічим не обмежена. Тобто, дослідник прикладної задачі самостійно планує порядок їх застосування.

Система Рівеста-Шаміра-Адлемана (RSA) – криптосистема з відкритим ключем, стійкість якої обумовлена складністю розкладу великого натурального числа на прості співмножники. Суть її полягає у наступному. Особа A обирає пару простих чисел p та q, $p \neq q$ і обчислює $n = pq$ та $\varphi(n) = (p-1)(q-1)$, де φ – функція Ойлера. Далі обирає число e, менше за $\varphi(n)$ та взаємно просте з $\varphi(n)$, і обчислює $d = e^{-1} \bmod \varphi(n)$. Пару (e, n) називають відкритим ключем особи A. Особа B може зашифрувати повідомлення для A, скориставшись ключем (e, n). Тільки особа A може відкрити шифротекст, оскільки тільки вона володіє таємним ключем (d, n). До чисел p і q, окрім довжини їх десяткового запису (порядку 150–200 розрядів), існує ряд інших вимог, як-от: це прості Блюма, $(p-1)/2$, $(q-1)/2$ – прості, різниця між ними більша за деяку граничну величину та ін.

Задачею факторизації називається наступна задача. Відомо, що $n \in N$ – складене. Знайти всі його прості дільники. Маємо частинний випадок цієї задачі – n є добутком двох простих чисел, але яких, ми не знаємо. На сьогодні не відомі поліноміальні алгоритми розв'язання цієї задачі. Експоненціальні алгоритми, наприклад, ті, що реалізують метод Ферма чи методи Поларда в деяких випадках працюють достатньо швидко, і їх можливо застосувати на практиці.

Розглянемо задачу обчислення простого дільника числа $n \in N$, $НСД(n, 2) = 1$ на множині

$$X = X(a,b;n) = \left\{ x : a \le x \le b,\ 2 < a < b \le \left\lceil \sqrt{n} \right\rceil, \right.$$
$$\left. \text{Í ÑÄ}(x,2) = 1,\ x \in N \right\}$$

Очевидно, $\forall d \in X$ і такого, що $d \mid n$ достатньо перевірки на простоту. Таку перевірку можна провести за поліноміальний час. Якщо знайдений дільник d не є простим, то $n \leftarrow n/d$ і в результаті, звужу-ючи множину X, отримаємо послідовність задач, перевіряючи при цьому коректність параметрів a і b. Тобто, залишається розв'я-зати простішу задачу обчислення дільника чила n на множині $X(a,b;n)$, якщо він там є. Сформулюємо цю задачу, як задачу комбіна-торної оптимізації (S, f, Σ). Параметрами задачі є M – число мурах, задіяних у побудові кандидатів у розв'язки (маршрутів) $s \in S$, m – довжина маршруту, яка одно-часно визначає обмеження з Σ.

Пошуковим простором для мурах є скінченна множина $V = \{v_i : v_i \in X, i = 1, \ldots, |X|\}$. Мураха є рандомізованим конструктивним алгоритмом, який будує розв'язок задачі оптимізації на повному графі $G = (V, E)$, де з ребрами $e_{ij} \in E$ можна асоціювати феромонний слід τ_{ij}, відображаючи довготривалу пам'ять колонії про попередній пошук. Також з ребрами можна асоціювати значення евристики ω_{ij} що відобра-жає початкове значення феромону (на вході задачі). Позначимо через t момент часу (ітерацію). Визначимо *ймовірність переходу* мурахи k з вершини v_i у вершину v_j на ітерації t в допустимому околі O_i^k:

$$p_{ij}(t) = \frac{\tau_{ij}(t)}{\sum_{O_i^k} \tau_{ij}(t)},$$

де $\tau_{ij}(t)$ - рівень феромону на ребрі e_{ij} в момент t.

У [25] вивчається фітнес-функція $f(x,y) = |n - xy|$, але з точки зору складності пошукового простору краще розглянути іншу.

Фітнес функцію f визначимо для маршруту s:

$$\vec{f}(s) = \begin{pmatrix} \sum_{i:\, v_i \in s} \{n/v_i\} \\ \min_{i:\, v_i \in s} \{n/v_i\} \end{pmatrix}.$$

Цілком природньо виглядає евристичне припущення, що дільник d числа n знаходиться на маршруті s_{k_0}, якщо $f_1(s_{k_0}) = \min_{1 \le k \le M} f_1(s_k)$. При цьому, можна припустити, що $d = \arg\min f_2(s_{k_0})$.

Далі, нехай Q - параметр, корегуючий зменшення відкладеного феромону, яке необхідно відбудеться із зростанням довжини маршруту, параметр ρ - інтенсивність випаровування, яку часто обирають рівною 0,6. Тоді, приріст феромону на ребрі e_{ij} маршруту s на ітерації t:

$$\Delta\tau_{ij}(t) = \frac{Q}{f_1(t)}.$$

Оновлення феромону з урахуванням випаровування:

$$\tau_{ij}(t+1) = (1-\rho)\left(\tau_{ij}(t) + \sum_{n} \Delta\tau_{ij,n}(t)\right),$$

де n - число мурашок, які пройшли ребро e_{ij}.

Доцільно передбачити евристичну оцінку θ_i асоційовану з вершиною графа v_i - числа агентів, які побували у вершині v_i. По досягненню її порогового значення (задається як параметр задачі), відповідну їй вершину слід виключати з множини пошуку як не перспективну, тим самим, звужуючи простір пошуку. Відповідно, слід передбачити таку структуру даних, яка забезпечить динамічну підтримку двох неперетинних підмножин множини X - перспективних і не перспективних точок для пошуку.

Для побудови розв'язку (наближеного розв'язку) агенти реалізують жадібну рандомізовану конструктивну евристику. Конструктивні алгоритми характеризуються простотою обчислювальної схеми і високою швидкістю. Приклад жадібної конструктивної евристики наведено на лістингу 2.

Лістинг 2:

```
ЖАДІБНА_КОНСТРУКТИВНА_ЕВРИСТИКА
    s_part ← ∅
    while ¬ умова завершення do
            e ← GreedyComponent(s_part)
            s_part ← s_part ∪ e
    return s_part
```

Функція *GreedyComponent* на кожному кроці повертає компоненту розв'язку з найкращою евристичною оцінкою. У найпростішому випадку, додається компонента розв'язку, обрана випадковим чином (рівноймовірно). Наприклад, у АСО до часткового розв'язку додається компонента, обчислена на підставі наявної евристичної інформації та феромонного сліду, який змінюється динамічно і відображає попередній пошуковий досвід колонії.

На цьому етапі уже достатньо інформації для розробки ітераційної обчислювальної процедури пошуку дільника на множині. Ймовірним покращенням такого алгоритму може бути додавання процедури пошуку в околі шляхів – кандидатів у (суб)оптимальний розв'язок. Псевдокод процедури локального пошуку наведено на лістингу 3.

Лістинг 3:

```
ІТЕРАТИВНЕ_ПОЛІПШЕННЯ( p ∈ P )
    p' ← Improve(p)
    while p' ≠ p do
            p ← p'
            p' ← Improve(p)
    return p
```

Функція *Improve* повертає поліпшений розв'язок з околу, якщо такий розв'язок існує, інакше, повертає поточний розв'язок і локальний пошук припиняється.

Застосування операцій генетичного алгоритму для *Improve* – може бути одним із способів локального поліпшення розв'язків, побудованих мурахами. Дійсно, нехай M – число мурашок, $S(t) = (s_1, s_2, \ldots, s_M)$ – відповідна їм множина маршрутів (популяція маршрутів) на ітерації t. Визначимо генетичні операції над $S(t)$.

CROSS. З множини маршрутів $(s_1, s_2, \ldots, s_M)$ утворюють пари за деяким критерієм Θ (наприклад, всі можливі пари маршрутів). Далі, за деяким критерієм Ψ від кожного компонента пари обирають представника (наприклад, випадковим чином) для *кросоверу* - операції ГА. Хромосомою генетичного алгоритму є двійкове зображення цілого числа – вершини маршруту. Нащадки, утворені в результаті кросинговеру можуть стати вершинами нових шляхів у популяції маршрутів, а можуть зайняти місця батьків як у канонічному ГА без розширення популяції маршрутів. Очевидно, результатом операції кросинговеру ГА можуть стати хромосоми, які не належатимуть області визначення нашої задачі. Тому в операції *CROSS* слід передбачити і перевірку області визначення і додаткові дії, що не дозволять вийти за область визначення задачі.

MUT. Перетворення (м*утація*) потомків із заданою ймовірністю p, шляхом інверсії генів у випадково обраних позиціях. Перевірку області визначення потрібно реалізувати і для даної операції.

SELECT. Відбір (*селекція*) маршрутів за деяким критерієм Ω для формування нового покоління і скорочення популяції.

Серйозною проблемою функції локального пошуку, як і генетичного алгоритму в цілому, є схильність приводити до локального оптимуму задачі. Таким чином, рішення про необхідність включення локального пошуку має бути зваженим.

Умовою зупинки обчислення дільника числа на множині, може бути настання хоча б однієї з подій: перевищення максимального числа ітерацій $t_{\max}$; для малого $\delta > 0$ існує k_0 такий, що $f_1(s_{k_0}) < \delta$; для малого $\gamma > 0$ існує k_0 такий, що $f_2(s_{k_0}) < \gamma$; критичне звуження простору пошуку. Результатом є (суб)оптимальний шлях s_{k_0} на момент зупинки, при цьому $d = \arg\min f_2(s_{k_0})$.

Висновки

Евристичні алгоритми дозволяють знаходити задовільні рішення складної проблеми за прийнятний час без необхідності теоретичного обгрунтування їх правильності та оптимальності. Задачі криптоаналізу є складними уже за визначенням. Методи, що базуються на виснов-ках з результатів спостережень над процесами, які відбуваються у живій та неживій природі, визначають клас метаевристичних алгоритмів. Насправді, такі алгоритми є стратегіями, які управляють ефективним досліженням простору пошуку. В результаті пошуку накопичуться нові знання, які автоматично враховуються на наступних етапах, тобто відбувається направ-лений інтелектуальний пошук. Результати інте-лектуального пошуку можна перевірити в рамках конкретної практичної задачі, хоча самі метаевристики не є проблемно орієнтованими.

Численні спроби атак, в тому числі успішних, класичних та деяких сучасних потокових шифрів із застосуванням метаевристик описані у літературі. Складність криптаналізу RSA обумовлена складністю розв'язання задачі факторизації. Відомі на сьогодні алгоритми факторизації мають (суб)експоненціальну складність і їх застосовність є обмеженою. Враховуючи акту-альність RSA, дослідження її криптостійкості, в тому числі метвевристичними методами, мають місце і носять закритий характер. Декілька поси-лань на дослідження цього типу можна знайти у огляді літератури. У дослідженні розглядаються особливості застосування ACO до даної задачі криптоаналізу.

ЛІТЕРАТУРА

[1] Гуляницький Л. Ф., Мулеса О. Ю. Прикладні методи комбінаторної оптимізації. Київ. 2016. 142 с.

[2] Ho Yean Li, A. S. Heuristic cryptanalysis of classical and modern ciphers. *2005 13th IEEE International Conference on Networks Jointly held with the 2005 IEEE 7th Malaysia International Conf on Communic*. 2005. *2*. 710–715. https://doi.org/10.1109/ICON.2005.1635595

[3] Toemeh R., Arumugam S. Breaking Transposition Cipher with Genetic Algorithm. *Elektronika Ir Elektrotechnika*. 2007. 79(7). 75-78. https://eejournal.ktu.lt/index.php/elt/article/view/10844

[4] Song, J.; Zhang, H.; Meng, Q.; Zhangyi, W. Cryptanalysis of Four-Round DES Based on Genetic Algorithm. *Wirel. Commun. Netw. Mob. Comput. IEEE*. 2007. 10. 2326–2329. https://doi.org/ 10.1109/WICOM.2007.580

[5] Garg P. A Comparison between Memetic algorithm and Genetic algorithm for the cryptanalysis of Simplified Data Encryption Standard algorithm. *Int. J. Netw. Secur. Its Appl. (IJNSA)*. 2009. 1. 34–42. https://doi.org/10.48550/arXiv. 1004.0574

[6] Hu W. Cryptanalysis of TEA using quantum-inspired genetic algorithms. *J. Softw. Eng. Appl*. 2010. 3. 50–57. http://dx.doi.org/10.4236/jsea.2010.31006

[7] Abd-Elmonim W. G., Ghali N. I., Hassanien A. E., Abraham. A. Known-Plaintext Attack of DES16 Using Particle Swarm Optimization. *In Proceedings of the Third IEEE World Congress on Nature and Biologically Inspired Computing, Salamanca, Spain*. 2011. 12–16. https://doi.org/10.1109/NaBIC.2011.6089410

[8] Mekhaznia T., Menai M. Cryptanalysis of classical ciphers with ant algorithms. *International Journal of Metaheuristics*. 2014. 3(3). 175-198. https://doi.org/10.1504/IJMHEUR.2014.065159

[9] Boryczka U., Dworak K. Genetic Transformation Techniques in Cryptanalysis. In: Nguyen, N. T., Attachoo, B., Trawiński, B., Somboonviwat, K. (eds) *Intelligent Information and Database Systems. ACIIDS 2014. Lecture Notes in Computer Science*. 2014. vol. 8398. Springer, Cham. https://doi.org/10.1007/978-3-319-05458-2_16

[10] Dadhich A., Gupta A., Yadav S. Swarm Intelligence based linear cryptanalysis of four-round Data Encryption Standard algorithm. In *2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*. 2014. 378–383. http://dx.doi.org/10.1109%2 FICICICT.2014.6781312

[11] Sadeghzadeh M, Taherbaghal M. A new method for decoding an encrypted text by genetic algorithms and its comparison with tabu search and simulated annealing.Management Science Letters. 2014. 4(2). 213–220. https://doi.org/10.5267/ j.msl.2013.12.037

[12] Teytaud F., Fonlupt, C. A Critical Reassessment of Evolutionary Algorithms on the cryptanalysis of the simplified data encryption standard algorithm. 2014. ArXiv, abs/1407.1993. https:// doi.org/10.5121/ IJCIS.2014.4201

[13] Ashok K. Bhateja, Aditi Bhateja, Santanu Chaudhury, P.K. Saxena, Cryptanalysis of Vigenere cipher using Cuckoo Search. *Applied Soft Computing*. 2015. Vol. 26. 315-324. https://doi.org/10.1016/j.asoc.2014.10.004

[14] Tahar, M. BAT algorithm for Cryptanalysis of Feistel cryptosystems . *International Journal of Intelligent Systems and Applications in Engineering*. 2015. 3(2). 82–85. https://doi.org/10.18201/ ijisae.82426

[15] Dworak K., Boryczka U. Differential Cryptanalysis of FEAL4 Using Evolutionary Algorithm. In: Nguyen, N., Iliadis, L., Manolopoulos, Y., Trawiński, B. (eds) *Computational Collective Intelligence. ICCCI 2016. Lecture Notes in Computer Science*. 2016.

vol. 9876. Springer, Cham. https://doi.org/10.1007/978-3-319-45246-3_10

[16] Amic S., Soyjaudah K.S., Mohabeer H., Ramsawock. G. Cryptanalysis of DES16 using binary firefly algorithm. In Proceedings of the *2016 IEEE International Conference on Emerging Technologies and Innovative Business Practices for the Transformation of Societies, Balaclava, Mauritius, 3–6 August 2016; IEEE: Balaclava, Mauritius*. 2016. 94–99. https://doi.org/10.1109/EmergiTech.2016.7737318

[17] Amic S., Soyjaudah K.S., Ramsawock G. Binary cat swarm optimization for cryptanalysis. In Proceedings of the *2017 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), Bhubaneswar, India*. 2017. 1–6. https://doi.org/10.1109/ANTS.2017.8384120

[18] Candra A., Budiman M. A., Rachmawati D. On Factoring The RSA Modulus Using Tabu Search. *Journal of Computing and Applied Informatics*. 2017. vol. 1. n. 1. 30–37. https://doi.org/10.32734/JOCAI.V1.I1-65

[19] Grari H., Azouaqui A., Zine-Dine K., Bakhouya M., Gaber J. Cryptanalysis of Knapsack Cipher Using Ant Colony Optimization. *Smart Application and Data Analysis for Smart Cities*. 2018. http://dx.doi.org/10.2139/ssrn.3185322

[20] Grari H., Azouaoui A., Zine-Dine K. A Novel Ant Colony Optimization Based Cryptanalysis of Substitution Cipher. In: Abraham, A., Haqiq, A., Ella Hassanien, A., Snasel, V., Alimi, A. (eds) Proceedings of the *Third International Afro-European Conference for Industrial Advancement -AECIA*-2016. *Advances in Intelligent Systems and Computing*. 2016. vol. 565. 180–187 https://doi.org/10.1007/978-3-319-60834-1_19

[21] Grari H., Azouaoui A., Zine-Dine K. Ant colony optimization for cryptanalysis of simplified-DES. In *Advanced Intelligent Systems for Sustainable Development (AI2SD'2018) Vol 2: Advanced Intelligent Systems Applied to Energy*. 2019. 111-121. https://doi.org/10.1007/978-3-030-12065-8_11

[22] Grari H., Azouaoui A., Zine-Dine K. A cryptanalytic attack of simplified-AES using ant colony optimization. *International Journal of Electrical & Computer Engineering*. 2019. 9(5). 4287-4295. https://doi.org/10.11591/ijece.v9i5.pp. 4287–4295

[23] Jain A., Chaudhari N. S. An improved genetic algorithm and a new discrete cuckoo algorithm for solving the classical substitution cipher. *International Journal of Applied Metaheuristic Computing (IJAMC)*. 2019. 10(2), 109-130. DOI: 10.4018/IJAMC.2019040105

[24] Amic S., Soyjaudah K.S., Ramsawock G. Dolphin swarm algorithm for cryptanalysis. In Information Systems Design and Intelligent Applications; Satapathy, S., Bhateja, V., Somanah, R., Yang, X.S., Senkerik, R., Eds.; *Advances in Intelligent Systems and Computing*.2019. Vol. 863. 149–163. https://doi.org/10.1007/978-981-13-3338-5_15

[25] D. Rachmawati, H. Tamara, S. Sembiring, M. Budiman. RSA public key solving technique by using genetic algorithm. *Journal of Theoretical and Applied Information Technology*. 2020. Vol. 98. No. 15. 2990-2999. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85053419786&partnerID=40&md5=1072f49ab20414f2288933fefbef056e

[26] Sabonchi A. K. S., Akay B. Cryptanalysis of polyalphabetic cipher using differential evolution algorithm. *Tehnički vjesnik*. 2020. 27(4). 1101-1107. https://doi.org/10.17559/TV-20190314095054

[27] Akay B. A binomial crossover based artificial bee colony algorithm for cryptanalysis of polyalphabetic cipher. *Tehnički vjesnik*. 2020. 27(6). 1825-1835. https://doi.org/10.17559/TV-20190422225110

[28] Sabonchi A. K. S., Akay B. A survey on the Metaheuristics for Cryptanalysis of Substitution and Transposition Ciphers. *Computer Systems Science And Engineering*. 2021. vol. 39. no. 1. 87-106. http://doi.org/10.32604/csse.2021.05365

[29] Grari H., Lamzabi S., Azouaoui A., Zine-Dine K. Cryptanalysis of Merkle-Hellman cipher using ant colony optimization. *Int J Artif Intell*. 2021. 490–500. DOI: 10.11591/ijai.v10.i2

[30] Dworak K., Boryczka U. Breaking Data Encryption Standard with a Reduced Number of Rounds Using Metaheuristics Differential Cryptanalysis. *Entropy*. 2021. vol. 23. no. 12: 1697. https://doi.org/ 10.3390/e23121697

[31] Jain A., Sharma P.C., Vishwakarma S.K., Gupta N.K., Gandhi V.C. Metaheuristic Techniques for Automated Cryptanalysis of Classical Transposition Cipher: A Review. In: Somani, A.K., Mundra, A., Doss, R., Bhattacharya, S. (eds) *Smart Systems: Innovations in Computing. Smart Innovation, Systems and Technologies*. 2022. vol. 235. https://doi.org/10.1007/978-981-16-2877-1_43

Кубайчук О. О.

ОСОБЛИВОСТІ ЗАСТОСУВАННЯ АЛГОРИТМУ ACO ДО ДЕЯКИХ ЗАДАЧ КРИПТОАНАЛІЗУ

Вимоги до інформаційної безпеки диктують неохідність розвитку нових методів криптоаналізу. Сучасний криптоаналіз спирається на математику, зокрема на теорію та методи оптимізації. Враховучи загальновизнані вимоги до зламостійкості шифрів, задача розшифрування мусить розглядатися, як задача комбінаторної оптимізації.

В роботі обґрунтовується необхідність розвитку нових методів криптоаналізу із застосуванням метаевристик, міститься ретрспективний огляд публікацій за останній період в даній області. Кількість публікацій свідчить про актуальність напрямку досліджень.

Розглядаються особливості застосування алгоритму ACO (Ant Colony Optimization) до задач криптоаналізу, зокрема, задачі факторизації. Описується структура і загальні принципи роботи алгоритму ACO, адаптація даного алгоритму до розв'язання конкретної задачі комбінаторної оптимізації. Розглянуто різні варіанти фітнес-функції, особливості їх застосування, способи звуження простору пошуку, правила вибору напрямку руху на графі, модифікація локального пошуку. Як один із варіантів модифікації розглядається додавання генетичних операторів кросоверу, мутації, селекції. Описано умови припинення роботи алгоритму.

Обґрунтовано доцільність застосування метаевристик для розв'зання задач комбінаторної оптимізації що виникають у різних предметних областях, зокрема, у криптоаналізі. Підкреслюється, що так як теоретичні дослідження алгоритмів комбінаторної оптимізації рідко дозволяють отримувати результати, які можуть бути застосовані на практиці, то основним інструментом аналізу їх ефективності є обчислювальний експеримент.

Ключові слова: криптоаналіз; ACO; оптимізація; евристика; метаевристика; фітнес-функція.

Kubaychuk O.

SPECIALITIES OF THE APPLICATION OF THE ACO ALGORITHM TO SOME CRYPTANALYSIS PROBLEMS

Requirements for information security dictate the necessity of developing new methods of cryptanalysis. Modern cryptanalysis depend on mathematics, in particular on theory and optimization methods. Taking into account the generally recognized requirements for attack resistance of ciphers, the decryption problem should be considered as a combinatorial optimization problem

The paper proves the necessary of the development of new methods of cryptanalysis using metaheuristics, contains a retrospective review of publications in the last period in this area. The number of publications indicates the relevance of the research direction.

Specialities of the application of the Ant Colony Optimization algorithm to cryptanalysis problems, in particular, factorization problem, are considered. The structure and general principles of the ACO algorithm are described, as well as the adaptation of this algorithm to the solution of a specific problem of combinatorial optimization. Various variants of the fitness function, features of their application, methods of narrowing the search space, rules for choosing the direction of movement on the graph, modification of local search are discussed. The addition of genetic operators of crossover, mutation, and selection is considered as one of the modification options. The conditions for stopping the operation of the algorithm are described.

The various facts of using metaheuristics for solving combinatorial optimization problems arising in numerous subject areas, in particular, in cryptanalysis, are described. It is emphasized that since theoretical studies of combinatorial optimization algorithms rarely allow obtaining results that can be applied in practice. The main tool for analyzing their effectiveness is a computational experiment.

Keywords: cryptanalysis; ACO; optimization; heuristics; metaheuristics; fitness function.

Стаття надійшла до редакції 27.02.2023 р.
Прийнято до друку 01.06.2023 р.