УДК 004 DOI https://doi.org/10.35546/kntu2078-4481.2024.1.37

М. В. ОНАЙ
кандидат технічних наук, доцент,
доцент кафедри програмного забезпечення комп'ютерних систем
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
ORCID: 0000-0002-4938-8355

Д. Т. ГУЛЬКО
аспірант кафедри програмного забезпечення комп'ютерних систем
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
ORCID: 0009-0008-6810-737X

КЛАСИФІКАЦІЯ МЕТОДІВ ДИСКРЕТНОГО ЛОГАРИФМУВАННЯ НА ЕЛІПТИЧНІЙ КРИВІЙ

У цій роботі проведено аналіз та класифікацію методів розв'язання задачі дискретного логарифмування у мультиплікативних та адитивних групах, а також обґрунтовано актуальність такого аналізу. Особливий акцент зроблено на розв'язанні цієї задачі на еліптичних кривих над скінченними полями. Робота спрямована на підвищення стійкості криптографічних систем шляхом аналізу та класифікації існуючих методів вирішення задачі дискретного логарифмування.

У статті розглянуто такі методи: метод перебору, метод Поліга-Геллмана, метод Деніела Шенкса та його модифікації, а саме метод Кенгуру та метод "Two Grumpy Giants and a Baby". Окрім того, у роботі розглянуто ρ-метод Полларда та його модифікацію, що передбачає розпаралелення на декілька потоків виконання, а також метод Лас-Вегаса – сучасний метод, що використовує матричні обчислення для розв'язання задачі дискретного логарифмування.

Ключовим аспектом цієї статті є комплексний порівняльний аналіз методів дискретного логарифмування, результати аналізу наведено у відповідних таблицях, де подана їх часова та просторова складність, а також низка інших показників. Проведений аналіз надає інформацію про ефективність, безпеку та практичність кожного методу, закладає основу для подальших досліджень, а також дозволяє будувати більш стійкі криптосистеми.

Визначено, що ρ-метод Полларда має найкращий баланс між швидкодією та пам'яттю, що використовується, тому висунуто гіпотези щодо його покращення. Перша гіпотеза полягає у тому, що при перевірці на існування колізії на кожній ітерації алгоритму, що реалізує цей метод, доцільно порівнювати не точки, а їх класи еквівалентності. Друга гіпотеза покращення полягає у скороченні інтервалу, в якому знаходиться колізія. Іншим перспективним методом вирішення задачі дискретного логарифмування є метод Лас-Вегаса, що має високу швидкодію, проте цей метод не гарантує рішення і має високу просторову складність.

Ключові слова: *проблема дискретного логарифмування, криптографія, криптостійкість, еліптична крива, метод Шенкса, ρ-метод Полларда, метод Поліга-Гелмана, метод Лас-Вегаса, скінченні поля, поля Галуа.*

M. V. ONAI
Candidate of Technical Sciences, Associate Professor,
Associate Professor at the Department of Computer Systems Software
National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute"
ORCID: 0000-0002-4938-8355

D. T. HULKO
Postgraduate Student at the Department of Computer Systems Software
National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute"
ORCID: 0009-0008-6810-737X

CLASSIFICATION OF METHODS FOR SOLVING THE ELLIPTIC CURVE DISCRETE LOGARITHM PROBLEM

In this work, the analysis and classification of methods for solving the discrete logarithm problem in multiplicative and additive groups is carried out, as well as the relevance of such an analysis is substantiated. Special emphasis is placed

on solving this problem on elliptic curves over finite fields. The work is aimed at increasing the stability of cryptographic systems by analyzing and classifying existing methods for solving the discrete logarithm problem.

The following methods are considered in the article: the brute force method, the Pohlig-Hellman method, the Daniel Shanks method and its modifications, namely the Kangaroo method and the "Two Grumpy Giants and a Baby" method. In addition, the work considers Pollard's rho method and its modification, which involves parallelization in several execution threads, as well as the Las Vegas method – a modern method that uses matrix calculations to solve a discrete logarithm problem.

The key aspect of this article is a comprehensive comparative analysis of discrete logarithm methods, the results of the analysis are given in the corresponding tables, where their temporal and spatial complexity, as well as a number of other indicators, are presented. The conducted analysis provides information on the effectiveness, security and practicality of each method, lays the foundation for further research, and also allows building more stable cryptosystems.

Pollard's rho algorithm was found to have the best balance between speed and memory usage, and hypotheses were put forward to improve it. The first hypothesis is that when checking for the existence of a collision at each iteration of the algorithm, it is advisable to compare not the points, but their equivalence classes. The second hypothesis of improvement consists in shortening the interval in which the collision occurs. Another promising method for solving the discrete logarithm problem is the Las Vegas method, which has high performance, but this method does not guarantee a solution and has high spatial complexity.

Key words: *discrete logarithm problem, cryptography, crypto resistance, elliptic curve, Shanks' method, Pollard's rho method, Pohlig-Hellman method, Las Vegas method, finite fields, Galois fields.*

Постановка проблеми

У сучасному світі, де цифрові технології стали невід'ємною частиною нашого повсякденного життя, захист конфіденційності даних та боротьба з кіберзагрозами стали важливими завданнями. В цьому контексті, дослідження у галузі криптографії набувають особливого значення, оскільки вони визначають принципи та методи захисту цифрових інформаційних потоків.

Перш за все, криптографія відіграє ключову роль у забезпеченні конфіденційності даних. Ця наука дозволяє створювати та використовувати алгоритми шифрування, які забезпечують захист інформації під час передачі даних через мережі або зберіганні в базах даних. Розвиток нових криптографічних методів є необхідним для забезпечення безпеки особистих даних, фінансових транзакцій та конфіденційної інформації.

Дослідження в галузі криптографії сприяють інноваціям та технологічному прогресу. Нові алгоритми та методи захисту від нових загроз розвиваються завдяки постійним дослідженням. Це сприяє появі нових технологій, які забезпечують безпеку в мережах, фінансових транзакціях та багатьох інших аспектах цифрової економіки.

Окрім того, для боротьби з кіберзлочинністю актуальним є як розроблення криптографічних алгоритмів та методів шифрування, так і постійний пошук вразливостей уже існуючих криптографічних систем та підходів, що дозволяє будувати нові системи з підвищеним рівнем захисту інформації від несанкціонованого доступу.

Розрізняють симетричне та асиметричне шифрування. Криптографічна стійкість протоколів асиметричного шифрування ґрунтується на використанні односторонньої функції. Однією з таких функцій є піднесення до степеня за модулем. Відповідно криптографічна стійкість таких систем залежить від складності задачі дискретного логарифмування.

Задача дискретного логарифмування полягає у знаходженні степеня, до якого потрібно піднести генератор циклічної скінченної групи, щоб отримати заданий елемент цієї групи. Ця група може бути як мультиплікативною, як запропоновано початково, так і адитивною. Так, наприклад, Кобліц [1] та Міллер [2] запропонували розглядати еліптичну криву у контексті цієї задачі, так як скалярне множення точки еліптичної кривої є піднесенням до степеня у адитивній групі. Іншими словами, задача дискретного логарифмування на еліптичній кривій полягає у тому, щоб визначити, скільки разів необхідно додати одну точку кривої саму до себе, щоб отримати іншу задану точку цієї ж кривої. Ця задача є важко вирішуваною, завдяки чому така криптографічна система є безпечною та використовується у різноманітних комунікативних протоколах (SSL/TLS, SSH, IPsec, тощо), блокчейні та криптовалютах, мобільних пристроях, IoT (Internet of things), тощо.

Проте існує ряд методів для вирішення задачі дискретного логарифмування, і нові методи продовжують досліджувати зараз. Ця проблема є актуальною, адже ці методи становлять загрозу для сучасних криптографічних систем, а їх оцінка та аналіз допомагають покращувати безпеку даних та уникати знайдених вразливостей, що використовуються у цих методах та алгоритмах.

Формулювання мети дослідження

Метою роботи є аналіз та класифікація методів дискретного логарифмування у мультиплікативних та адитивних групах, що дозволить побудувати більш стійкі до атак криптографічні системи.

Постановка задачі дискретного логарифмування

Сформулюємо проблему дискретного логарифму для еліптичної кривої математично. Нехай $E(\mathbb{F}_q)$ є еліптичною кривою, де $q = p^n, p$, – це просте число, $n \in \mathbb{N}$a. Очевидно, що абелева група $E(\mathbb{F}_q)$ не є обов'язково циклічною, проте вона має циклічну підгрупу достатньо «великого» порядку N . Надалі

будемо використовувати позначення $E(\mathbb{F}_q)$ як позначення цієї циклічної підгрупи. Нехай $P, Q \in E(\mathbb{F}_q)$, – точки на еліптичній кривій, такі що $kP = Q$. Задача дискретного логарифмування полягає у знаходженні скаляра k, що задовольняє цю рівність.

Задача дискретного логарифмування, на якій будується сучасна асиметрична криптографія, не має ефективного рішення. Це означає, що обчислювальної потужності сучасних комп'ютерних систем не вистачить для швидкого вирішення такої задачі.

Метод перебору

Перебір усіх можливих результатів додавання точки P самої до себе дозволяє обчислити дискретний логарифм за скінченний час. Проте такий підхід, очевидно, займе у найгіршому випадку N операцій, а складність такого алгоритму є $O(N)$. Зазвичай лінійна складність алгоритму не є високою, проте у контексті дискретного лоагрифмування в криптографії, значення N є настільки великим, що обчислення дискретного логарифму сучасними комп'ютерними системами може тривати роками. Мета вдосконалення будь-якого алгоритму вирішення задачі дискретного логарифмування полягає у покращенні часу виконання та затрат на використання пам'яті [3].

Метод Поліґа-Геллмана

Метод Поліґа-Геллмана є ефективним для груп з порядком, що є гладким числом, тобто таким, що не має великих простих дільників. Цей метод реалізується рекурсивним алгоритмом, що спрощує задачу шляхом обчислення дискретного логарифму у підгрупах простого порядку. Так, дискретний логарифм у кожній підгрупі може бути обчислений іншими методами, наприклад, методом Полларда.

Таким чином, основна ідея методу Поліґа-Геллмана полягає у спрощенні початкової задачі. Проте, для ефективного спрощення, як було сказано вище, необхідно, щоб порядок групи у задачі дискретного логарифмування, був гладким числом. При цьому вводиться додатковий параметр яким задається максимальне граничне значення простого множника, що отримується внаслідок факторизації порядку групи [4].

Нехай порядок групи може бути представлений як добуток:

$$N = \prod_{i=1}^{r} p_i^{c_i}, p_1 < p_2 < \cdots < p_r.$$

Для того щоб знайти рішення задачі дискретного логарифмування, тобто скаляр k, побудуємо систему рівнянь $k_i = k \ (mod\ p_i^{c_i})$ для всіх $1 \le i \le r$. Тут k_i є частковим рішенням задачі дискретного логарифмування. Ця система рівнянь може бути розв'язана, згідно Китайської теореми про остачі, якщо відомі часткові рішення k_i. Для того, щоб їх знайти необхідно обчислити c_i дискретних логарифмів у підгрупі $\langle P \rangle$ порядку p_i для кожного часткового рішення k_i.

Для пошуку часткових рішень представимо кожне k_i у системі числення за основою p_i, так що $k_i = z_0 + z_1 p_i + z_2 p_i^2 + \cdots + z_{c_i} p_i^{c_i - 1}$, і вирахуємо кожну цифру z_j послідовно. Визначимо початкові точки $P_0 = \frac{N}{p_i} P$ та $Q_0 = \frac{N}{p_i} Q$. Зрозуміло, що P_0 має порядок p_i, так як, $p_i P_0 = p_i \frac{N}{p_i} P = NP$ тоді:

$$Q_0 = \frac{N}{p_i} Q = \frac{N}{p_i} (kP) = kP_0 = z_0 P_0.$$

Для знаходження першої цифри z_0 необхідно вирішити задачу дискретного логарифмування для групи $\langle P_0 \rangle$. Після цього обчислюється $Q_1 = \frac{N}{p_i^2} (Q - z_0 P) = z_1 P_0$, і так далі. Загалом, якщо відомі цифри $z_0, z_1, \dots, z_{t-1}$, то для знаходження наступної цифри z_t необхідно вирішити задачу дискретного логарифмування $Q_t = z_t P_0$, де

$$Q_t = \frac{N}{p_i^{t+1}} \left(Q - \sum_{j=0}^{t-1} z_j p_i^j P \right).$$

Таким чином, задача дискретного логарифмування суттєво спрощується. Складність цього алгоритму можна визначити як $O(\sqrt{N})$ операцій групи у найгіршому випадку, проте, якщо порядок групи є гладким, то складність стає $O\left(\sum_i^r c_i (\log N + \sqrt{p_i})\right)$ при умові, що для пошуку часткових рішень використовується алгоритм зі складністю $O\left(\sqrt{p_i^{c_i}}\right)$ операцій групи.

Для захисту від атаки Поліґа-Геллмана рекомендується мати розмір групи кратний певному великому простому числу. Таким чином факторизація порядку групи не дасть особливого пришвидшення алгоритму.

Метод Шенкса та його модифікації

Метод Шенкса ще називають Baby step-Giant step (Маленький крок-Великий крок), як його назвав Деніел Шенкс [5]. Алгоритм, що реалізує цей метод може бути виконаний за детермінований час $O(\sqrt{N})$ операцій групи [6]. Також цей алгоритм вимагає збереження $\sqrt{N}$ елементів групи.

Перш ніж розглянути алгоритм Шенкса, варто зауважити, що шуканий скалярний множник k міститься у інтервалі $[1, N]$, і, як наслідок, k може бути представлений як декомпозиція $k = mq - r$, де , $m = \lceil\sqrt{N}\rceil$, $0 \le r < m$ та $0 \le q \le m$.

Для знаходження k ми створюємо набір «маленьких кроків» $\{ iP \}, 0 \le i \le m$. Маючи цей набір, починаємо обчислювати точки, роблячи «великий крок» $Q - jmP, 0 \le j \le m - 1$, поки не знайдемо співпадіння з набором «маленьких кроків». Якщо знайдено співпадіння $iP = Q - jmP$, то, враховуючи, що $kP = Q$, можемо обчислити:

$$k \equiv i + jm \ (mod\ N).$$

Якщо співпадінь не знайдено, то можна стверджувати, що не існує такого k , щоб задовільнити вхідні умови.

Цей алгоритм за найгіршого сценарію матиме швидкість виконання $2\sqrt{N}$ операцій групи, проте Поллард довів, що можливо покращити цю швидкість до $1.33\sqrt{N}$ операцій групи для довільних груп [7]. Це покращення називається методом Кенгуру. На відміну від методу Шенкса, Поллард пропонує одній сутності (кенгуру) робити один або декілька кроків випадкової довжини за одну ітерацію, а іншій (валлабі) робити лише один крок за ту ж ітерацію. Після кожного кроку робиться перевірка, чи не співпадають точки, на яких знаходяться обидві сутності. Якщо знайдена колізія, то можна обчислити дискретний логарифм.

Інше покращення базового алгоритму запропоновано Бернштейном та Ланге [8] під назвою Two Grumpy Giants and a Baby. Це покращення ставить за мету зменшити використання пам'яті базовим алгоритмом, при незмінній швидкості. Авторами цього методу вводяться три сутності, що обходять криву, починаючи з різних точок та з різним кроком. Так, якщо n_0 є певним маленьким цілим числом, а $M \approx 0.5\sqrt{N}$, то перша сутність робить кроки довжиною $n_0 P$ починаючи з точки P, наступна сутність робить кроки довжиною $P' = MP$ починаючи з точки Q, а остання сутність робить крок довжиною $P'' = -(M + 1)P$, починаючи з точки $2Q$. На відміну від оригінального методу Шенкса, у цій модифікації пропонується виконувати всі три кроки паралельно, зберігаючи проміжні результати у відповідний список та на кожній ітерації шукаючи співпадіння у цих списках. Варто зазначити, що чіткої оцінки швидкодії цього методу немає.

ρ-метод Полларда та його модифікації

Складність ρ-методу Полларда така сама, як і у методу Шенкса – $O(\sqrt{N})$ – проте він використовує сталий обсяг пам'яті $O(1)$, що робить його набагато ефективнішим з точки зору пам'яті, що використовується [9]. Так, наприклад, майже всі найбільші досягнення у вирішенні задачі дискретного логарифмування, що були зроблені та опубліковані компанією Certicom, були вирішені саме цим методом або його модифікаціями [10].

Перший крок алгоритму, що реалізує ρ-метод Полларда, полягає у розбитті еліптичної кривої на її попарно неперетинні підмножини $F_1, F_2, \dots F_s$, такі що $F_1 \cup F_2 \cup \dots \cup F_s = E(\mathbb{F}_q)$. Ці підмножини обираються випадковим чином, але вони мають бути приблизно однакового розміру. Рекомендована кількість підмножин s має бути приблизно рівною 20 [11]. На наступному кроці необхідно визначити гомоморфізм $f: E(\mathbb{F}_q) \to E(\mathbb{F}_q)$ як кусково-задану функцію $f(V) = V + a_i P + b_i Q$, де $V \in F_i$, а пари чисел a_i та b_i вибираються випадковим чином, так що $1 \le a_i, b_i \le N$. Далі необхідно випадковим чином вибрати початкову точку $P_0 = a_0 P + b_0 Q$, та почати обчислення

$$P_{i+1} = f(P_i).$$

Варто звернути увагу, що для кожної точки підтримується виконання рівності $P_i = u_i P + v_i Q$. У пам'яті необхідно зберігати триплет (P_i, u_i, v_i). Цей ітераційний процес проводиться для того, щоб знайти колізію, тобто співпадіння $P_i = P_j$, з якого можемо обчислити значення k :

$$k \equiv (v_j - v_i)^{-1}(u_i - u_j)\left(mod\ \frac{N}{\gcd(N, v_j - v_i)}\right).$$

Якщо зберігати кожен триплет у пам'яті, то використаний простір буде розширено до $O(\sqrt{N})$, тому пропонується розглянути метод одного триплету [12]. Початково у пам'яті зберігається триплет (P_0, u_0, v_0). Нехай на певній ітерації збережено триплет (P_i, u_i, v_i), тоді ми обчислюємо та порівнюємо з ним кожен триплет від $(P_{i+1}, u_{i+1}, v_{i+1})$ до $(P_{2i-1}, u_{2i-1}, v_{2i-1})$. Якщо знайдено співпадіння, то зупиняємо ітераційний процес. Якщо співпадінь не знайдено, то обчислюється та зберігається у пам'ять триплет (P_{2i}, u_{2i}, v_{2i}) для подальших ітерацій.

Необхідно зазначити, що цей метод засновано на теорії випадкових процесів, він належить до групи методів Монте-Карло. Поллард стверджує, що цей метод завжди знаходить співпадіння, так як ми обчислюємо нескінченну послідовність з P_i, що є елементами скінченної групи. Потрібно лише $O(\sqrt{N})$ елементів цієї послідовності, щоб ймовірність знайти співпадіння була більше за 0.5. Математично можна довести, що існує число $i \le 3\sqrt{N}$, для якого $P_i = P_{2i}$ [3]. Саме цим і визначається час виконання алгоритму для найгіршого випадку.

Модифікацією цього методу є підхід, що полягає у паралелізації процесу пошуку колізії та виконання ітераційного процесу одразу для декількох початкових точок. Нехай існують M потоків виконання, що незалежно

обирають початкову точку, P_0^i, де i – це номер потоку. Всі потоки використовують спільну функцію f обходу послідовності. Тоді якщо всі M потоків виконання будуть незалежними, то хоча б один з них знайде рішення приблизно за з$\sqrt{\frac{N}{M}}$ операцій групи. Проте якщо кожен потік виконання відправлятиме знайдені точки на спільний сервер для збереження та пошуку колізії, то очікувана кількість операцій групи на один потік виконання може буи оцінена як $\frac{1}{M}\sqrt{\frac{\pi N}{2}}$. Якщо відправляти на сервер всі знайдені точки, то затрати пам'яті будуть надзвичайно великими, тому натомість пропонується зберігати лише точки з певною специфічною властивістю. Специфічною точка може вважатись за умови, наприклад, що перші t бітів її x- координати дорівнюють нулю. Якщо частка специфічних точок у $\langle P \rangle$ дорівнює θ, то наступна специфічна точка після попередньої очікується за $\frac{1}{\theta}$ ітерацій. Таким чином, очікувана кількість операцій групи для знаходження колізії специфічних точок кожним незалежним потоком виконання є $\frac{1}{M}\sqrt{\frac{\pi N}{2}} + \frac{1}{\theta}$.

У випадку використання паралельного методу Полларда необхідно враховувати апаратне та програмне забезпечення, за допомогою якого проводиться виконання алгоритму. Залежно від нього потрібно знайти баланс між покращенням швидкодії та пам'яттю, що використовується.

Метод Лас-Вегаса

Окрім методів, що виконують пошук циклу у послідовності, існують методи, що базуються на матричному підході. Так, наприклад, один з найновіших підходів для вирішення задачі дискретного логарифмування є метод Лас-Вегаса [13]. Цей метод зводить задачу дискретного логарифмування до «проблеми L», тобто до задачі лінійної алгебри. Згідно цього методу необхідно побудувати матрицю M та обчислити її ліве ядро K. Матриця M складається з m рядків, де $m = 3n'$. та n' визначається плоскою проєктивною кривою $C = \sum_{u+v+w=n'} a_{u,v,w} x^u y^v z^w$, де $a_{u,v,w} \in \mathbb{F}_q$. Структура $x^u y^v z^w$ охоплює всі можливі одночлени степеня n', які, у свою чергу, є упорядкованими, що є важливим у подальшій побудові матриці M. Далі пропонується обирати випадкові незалежні цілі числа n_i, такі що $1 \le n_i \le N$. За допомогою них необхідно обчислити точки на еліптичній кривій $P_i = n_i P$. Матриця M будується шляхом додавання рядків до неї, відповідно до кожної точки $P_i = (x_i; y_i; z_i)$. У кожному рядку $(x_i; y_i; z_i)$ замінюється на відповідні x, y та z кривої C. Тепер необхідно знайти ліве ядро K матриці M. Якщо ж поглянути на праве ядро, то його елементи породжують криву C, тому вона очевидно проходить через всі точки P_i, з яких матриця M була побудована.

Якщо отримане ліве ядро K не є нульовим, кожен ненульовий вектор в K відповідає плоскій проєктивній кривій C степеня n' над полем $\mathbb{F}_q$. При чому ця крива не містить в собі криву $E(\mathbb{F}_q)$ та перетинає її у $3n'$ точок по одному разу. Ця кількість визначається теоремою Безу.

Введемо ще декілька позначень: l позначає кількість додаткових точок, які не входять у m точок перетину кривих. Введення таких додаткових точок дозволяє перевіряти одночасно до $\binom{m+l}{l}$ варіантів різних кривих C представлених групою точок перетину розміру m. Стверджується, що $l = m$ є найбільш оптимальним значенням для кількості додаткових точок. Якщо порядок N еліптичної кривої $E(\mathbb{F}_q)$ є простим числом, то найбільш ефективні значення цих чисел будуть $l = m = \log_2 N$.

Наступними обираються додатні цілі числа s та t, такі що $s \neq t$, але $s + t = m + l$. Ці числа використовуються для побудови набору точок $P_i = n_i P$, $i = 1,2, \dots, s$, та $Q_j = -n_j Q$, $j = 1,2, \dots, t$, з яких будується матриця M. За таких умов ліве ядро K буде мати розмірність l, і якщо існує вектор v в K з хоча б l нулів, то існує крива C, що проходить через $3n'$ точок, що відповідають ненульовим точкам вектора v. Пошук цього вектора називається "проблемою L".

Якщо порядок N еліптичної кривої $E(\mathbb{F}_q)$ є простим числом та прямує до нескінченності, то метод Лас-Вегаса для спрощення задачі дискретного логарифмування до задачі лінійної алгебри (проблеми L) має ймовірність успіху приблизно $1 - \frac{1}{e} \approx 0.6321$. Розмірність же матриці, що необхідна для досягнення такої ймовірності, дорівнює $O(\log N)$. Часова складність цього методу також може бути оцінена як $O(\log N)$.

Щодо подальшого розв'язку задачі, то вирішення проблеми L буде вирішенням задачі дискретного логарифмування. У перших роботах авторами методу пропонувалось використання методу Гауса та доповнення Шура [14], проте у новіших роботах пропонуються покращення шляхом використання методу початкових мінорів [15].

Порівняння методів

Порівнюючи методи варто зазначити, що для вибору найкращого методу, потрібно враховувати багато факторів, серед яких: потужність апаратного забезпечення, розмір вхідних даних, оптимізація програмного забезпечення, обмеження у часових та просторових параметрах тощо. У таблиці 1 наведено порівняльну характеристику описаних методів, де складність методів представлена у вигляді константи C, такій що алгоритм, який реалізує відповідний метод, виконується за $(c + o(1))$ операцій групи.

Таблиця 1

Порівняння складності методів

Метод	Середньостатистичний випадок	Найгірший випадок
Метод Поліга-Геллмана	$O\left(\sum_{i}^{r} c_i(\log N + \sqrt{p_i})\right)$	$2\sqrt{N}$
Метод Шенкса	$1.5\sqrt{N}$	$2\sqrt{N}$
Метод Кенгуру	$1.33\sqrt{N}$	$2\sqrt{N}$
Метод Two Grumpy Giants and a Baby	$1.25\sqrt{N}$	$2\sqrt{N}$
ρ-метод Полларда	$1.253\sqrt{N}$	До $3\sqrt{N}$
Паралельний ρ-метод Полларда у M потоках виконання	$\frac{1.253}{M}\sqrt{N}$	До $3\sqrt{N}$
Метод Лас-Вегаса	$6\log N$	∞

Як можна бачити з таблиці 1, найбільш перспективним з точки зору швидкодії є метод Лас-Вегаса, а наступним є ρ-метод Полларда. Метод Two Grumpy Giants and a Baby також має високу швидкодію, проте її значення підтверджено евристично, а не теоретично.

Складність виконання методу не може бути єдиною його характеристикою. У таблиці 2 наведена порівняльна характеристика методів за іншими показниками.

Таблиця 2

Порівняльна характеристика методів

Метод	**Просторова складність**	**Підхід**	**Гарантія вирішення задачі**	**Універсальність**
Метод Поліга-Геллмана	$O(\log N)$	Факторизація	Так	Для груп із «гладким» порядком
Метод Шенкса	$O(\sqrt{N})$	Пошук колізії	Так	Для всіх груп
Метод Кенгуру	$O(1)$	Пошук колізії	Так	Для всіх груп
Метод Two Grumpy Giants and a Baby	$O(\sqrt{N})$	Пошук колізії	Так	Для всіх груп
ρ-метод Полларда	$O(1)$	Пошук колізії	Так	Для всіх груп
Паралельний ρ-метод Полларда	$O(\sqrt{N})$	Пошук колізії	Так	Для всіх груп
Метод Лас-Вегаса	$O(\log N)$	Матричний підхід	Ні	Для еліптичних кривих над полем $GF(p^m)$

Слід зазначити, що метод Поліга-Геллмана та метод Лас-Вегаса не є методами вирішення задачі дискретного логарифмування, проте вони дозволяють її спростити. При цьому метод Поліга-Геллмана спрощує задачу до самої себе, у той час, коли метод Лас-Вегаса спрощує її до задачі лінійної алгебри.

Головним недоліком методу Лас-Вегаса є відсутність гарантії вирішення задачі.

Метод Полларда є універсальним, гарантує вирішення задачі та має сталі затрати на обсяг пам'яті, що використовується, тому є найбільш ефективним, враховуючи його швидкість Можна висунути наступну гіпотезу щодо його покращення: доцільно при перевірці на існування колізії порівнювати не точки, а їх класи еквівалентності. Класи еквівалентності пропонується визначати наступним чином: $P \sim P'$ якщо $P' \in \{P, -P\}$. Кількість елементів в одному класі еквівалентності дорівнює $2t + 2$, якщо t – кількість перетинів еліптичною кривою осі абсцис.

Окрім того, варто зазначити, що в основі методу Полларда лежить алгоритм Флойда знаходження циклу в послідовності. Згідно теореми, яку запропонував Флойд якщо у послідовності є цикл, то існує певне число m, для якого справедливо $x_m = x_{2m}$, при чому найменше значення m знаходиться в інтервалі, $T \leq m \leq T + M$, де M – довжина циклу, а T – індекс першого елементу цього циклу. За іншою теоремою, відомо, що для перемішаної випадковим чином послідовності буде справедливою рівність $T + M \approx \sqrt{\frac{N\pi}{2}} \approx 1.25\sqrt{N}$. На даний момент способа для знаходження нижньої границі цього інтервалу немає, тому це є актуальною задачею.

Висновки

Розглянуто методи для вирішення задачі дискретного логарифмування, що в середньому мають складність виконання $O(\sqrt{N})$. Так як ρ-метод Полларда використовує набагато менший обсяг пам'яті з приблизно схожою складністю виконання, порівняно з методом Шенкса, то пропонується проводити дослідження саме цього методу, приділивши також увагу новим, можливо, більш перспективним методам на основі матричного підходу. Запропоновано спосіб вдосконалення ρ-методу Полларда за рахунок спрощення процесу перевірки колізії шляхом введення класів еквівалентності. Окрім цього необхідно зосередити подальші дослідження на пошуку нижньої границі інтервалу, в якому може статись колізія. Якщо ця границя буде знайдена, можна буде пропустити перші T елементів послідовності, що зменшить час роботи алгоритму, який реалізує даний метод.

Список використаної літератури

1. Koblitz N. Elliptic curve cryptosystems. *Mathematics of Computation*. 1987. Vol. 19, No. 177, P. 203–209.

2. Miller V. Use of elliptic curves in cryptography. *Conference on the theory and application of cryptographic techniques*: Berlin, Heidelberg: Springer Berlin Heidelberg, August. 1987. Berlin, 1987. P. 417–426.

3. McCurley K. S., The discrete logarithm problem. *In Proc. of Symp. in Applied Math*. 1990. Vol. 42, P. 49–74.

4. Weisstein E. W. Smooth Number. URL: https://mathworld.wolfram.com/SmoothNumber.html.

5. Shanks D. Class number, a theory of factorization and genera. *In Proceedings of symposia Math. Soc*. 1971. Vol. 20, P. 415–440.

6. Galbraith S. D., Wang P., Zhang F. Computing elliptic curve discrete logarithms with improved baby-step giant-step algorithm. *ePrint Archive*. 2015.

7. Pollard J. Kangaroos, Monopoly and discrete logarithms. *Journal of cryptology*. 2000. Vol. 13, No. 4, P. 437–447.

8. Daniel J. Bernstein et. al. On the Correct Use of the Negation Map in the Pollard rho Method. *In Public Key Cryptography 2011*: 14th International Conference on Practice and Theory in Public Key Cryptography, Taormina, Italy, March 6–9. 2011. Taormina, 2011. Proc. 14, P. 128–146.

9. Pollard, J. M. Monte Carlo methods for index computation ($modp$). *Mathematics of computation*. 1978. Vol. 32, No. 143, P. 918–924.

10. Certicom Research, Certicom ECC Challenge. URL: https://www.certicom.com/content/dam/certicom/images/pdfs/challenge-2009.pdf

11. NKECK, J. L. The Discrete Logarithm Problem on Elliptic Curves Cryptography: Doctoral dissertation, Stellenbosch University, South Africa. 2014.

12. McConnachie, S. Pollard's rho-algorithm, and its applications to elliptic curve cryptography. 2007.

13. Abdullah, A., Mahalanobis, A., & Mallick, V. M. A new method for solving the elliptic curve discrete logarithm problem. *Journal of Groups, complexity, cryptology*. 2021. Vol. 12.

14. Mahalanobis, A., Mallick, V. M., & Abdullah, A. A Las Vegas algorithm to solve the elliptic curve discrete logarithm problem. *In Progress in Cryptology–INDOCRYPT 2018*: 19th International Conference on Cryptology in India, New Delhi, India, December 9–12. 2018. New Delhi, 2018. Proc. 19, P. 215–227.

15. Abdullah, A., Mahalanobis, A. Minors solve the elliptic curve discrete logarithm problem. *arXiv preprint arXiv:2310.04132*. 2023.

References

1. Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of computation*, vol. 48 no. 177, pp. 203–209.

2. Miller, V. S. (1985). Use of elliptic curves in cryptography. *In Conference on the theory and application of cryptographic techniques* (pp. 417–426). Berlin, Heidelberg: Springer Berlin Heidelberg.

3. McCurley, K. S. (1990). The discrete logarithm problem. *In Proceedings of symposia in Applied Math* (Vol. 42, pp. 49–74).

4. Weisstein, E. W. (n.d.). Smooth Number. Wolfram MathWorld. Retrieved from https://mathworld.wolfram.com/SmoothNumber.html.

5. Shanks, D. (1971). Class number, a theory of factorization, and genera. *In Proceedings of symposia Math. Soc*., 1971 (Vol. 20, pp. 415–440).

6. Galbraith, S. D., Wang, P., & Zhang, F. (2015). Computing elliptic curve discrete logarithms with improved baby-step giant-step algorithm. *Cryptology ePrint Archive.*

7. Pollard, J. M. (2000). Kangaroos, monopoly and discrete logarithms. *Journal of cryptology*, vol. 13, no. 4, pp. 437–447.

8. Bernstein, D. J., Lange, T., & Schwabe, P. (2011). On the correct use of the negation map in the Pollard rho method. *In Public Key Cryptography–PKC 2011*: 14th International Conference on Practice and Theory in Public Key Cryptography, Taormina, Italy, March 6–9, 2011. Proceedings 14 (pp. 128–146).

9. Pollard, J. M. (1978). Monte Carlo methods for index computation ($mod p$). *Mathematics of computation*, vol. 32, no. 143, pp. 918–924.

10. Certicom Research (2009). Certicom ECC Challenge. Retrieved from https://www.certicom.com/content/dam/certicom/images/pdfs/challenge-2009.pdf

11. NKECK, J. L. (2014). The Discrete Logarithm Problem on Elliptic Curves Cryptography (Doctoral dissertation, Stellenbosch University, South Africa).

12. McConnachie, S. (2007). Pollard's rho-algorithm, and its applications to elliptic curve cryptography.

13. Abdullah, A., Mahalanobis, A., & Mallick, V. M. (2021). A new method for solving the elliptic curve discrete logarithm problem. *In Journal of Groups, complexity, cryptology*, 12.

14. Mahalanobis, A., Mallick, V. M., & Abdullah, A. (2018). A Las Vegas algorithm to solve the elliptic curve discrete logarithm problem. *In Progress in Cryptology–INDOCRYPT 2018*: 19th International Conference on Cryptology in India, New Delhi, India, December 9–12, 2018, Proceedings 19 (pp. 215–227).

15. Abdullah, A., & Mahalanobis, A. (2023). Minors solve the elliptic curve discrete logarithm problem. *arXiv preprint arXiv:2310.04132*.