ТЕОРЕТИЧНІ ТА ПРИКЛАДНІ ПРОБЛЕМИ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ ПІДТРИМАННЯ ПРИЙНЯТТЯ РІШЕНЬ

УДК 511:003.26.09
DOI: 10.20535/SRIT.2308-8893.2018.1.08

ПРИСКОРЕННЯ МЕТОДУ КВАДРАТИЧНОГО РЕШЕТА НА ОСНОВІ ВИКОРИСТАННЯ УМОВНО *B*-ГЛАДКИХ ЧИСЕЛ

В.М. МІСЬКО

Анотація. Досліджено ступінь прискорення базового методу квадратичного решета на основі пошуку умовно *B*-гладких чисел. Проведено аналіз впливу на ефективність алгоритму та кількості випадків використання умовно *B*-гладких чисел. Показано, що модифікований алгоритм на основі пошуку умовно *B*-гладких чисел дозволяє факторизувати число у тих випадках, коли базовий алгоритм квадратичного решета (за стандартного інтервалу просіювання та розміру факторної бази) не зміг сформувати матрицю для отримання розв'язку.

Ключові слова: факторизація, метод квадратичного решета, умовно *B*-гладкі числа, прискорення.

ВСТУП

В інформаційно-телекомунікаційних системах для розв'язання задачі захисту інформації часто використовують RSA алгоритм. Поширення цього алгоритму робить актуальним його криптоаналіз. В основі криптостійкості найбільш популярного сьогодні асиметричного криптоалгоритму RSA є складність факторизації великих цілих чисел. Відкритий ключ містить велике складене ціле число — криптомодуль N, що є добутком двох великих простих чисел. Натепер немає відомого простішого універсального способу зламати шифрування як факторизація N. Тоді можемо отримати два прості числа з добутку та розшифрувати повідомлення [7, 8].

У 1977 р., коли був винайдений алгоритм RSA, факторизація цілих чисел з 80-десятковими знаками здавалась неможливою; 256-бітові ключі були надійними. Першим суттєвим проривом стало квадратичне решето (Quadratic Sieve) [1] — метод, винайдений Карлом Померансом у 1981 р., який може факторизувати числа розміром 100 десяткових символів і більше. Натепер це найефективніший відомий метод факторизації чисел, розміром меншим за 100 десяткових знаків. Поява ідей, які дозволять знизити обчислювальну складність методу квадратичного решета, може розширити множину великих чисел (більше 100 десяткових занків), де цей метод буде найкращим, це дасть змогу удосконалити процес криптоаналізу, хоча може призвести до збільшення числа розрядів N для криптостійких шифрів RSA.

© *В.М. Місько*, 2018

Тому розроблення нових способів прискорення методу квадратичного решета та їх дослідження є актуальним. У дослідженні пропонується додатково використовувати *B*-гладкі з неодиничними залишками, які є квадратами простих чисел. Такі *B*-гладкі числа будемо називати **умовно *B*-гладкими**.

ПОСТАНОВКА ЗАДАЧІ

Припустімо, що N — число, яке потрібно факторизувати, алгоритм квадратичного решета намагається знайти два числа x та y таких, щоб $x \neq \pm y(\mathrm{mod}\, n)$ і $x^2 = y^2(\mathrm{mod}\, n)$. Це означатиме, що $(x-y)(x+y) = 0(\mathrm{mod}\, n)$, тоді вирахуємо множники N як $\gcd(x-y,n)$ і $\gcd(x+y,n)$, використовуючи алгоритм Евкліда. Є імовірність 50 відсотків шансу, що цей додаток буде нетривіальним дільником N [2].

Алгоритм квадратичного решета генерує послідовність квадратів, використовуючи багаточлен $x^2 - N$, змінюючи x від $\sqrt{N}$ до $\sqrt{N} + M$ [2]. Величина M збільшується до межі $|M| \leq L^b$, де L^b — **інтервал просіювання**. Це місце, де метод стає евристичним, оскільки абсолютно точного способу обчислення інтервалу просіювання немає.

У квадратичному решеті вираховуємо залишок $x^2 \mathrm{mod}\, N$ для деяких x і знаходимо таку множину, добуток елементів якої є квадратом. Це дає змогу порівняти квадрати. Однак піднесення до квадрата множини випадкових чисел за модулем *N* приводить до великої кількості різних простих множників, великих векторів та до великого розміру матриці спеціальної системи лінійних рівнянь. Тому для спрощення додатково шукаємо пари цілих чисел *x* і *y(x)*, які відповідають значно простішим умовам ніж $x^2 = y^2 (\mathrm{mod}\, N)$. Алгоритм вибирає набір простих чисел, **який називається факторною базою**, і намагається знайти x таке, щоб залишок $y(x) = x^2 \mathrm{mod}\, N$ був добутком простих чисел, що входять до факторної бази. Такі x називаються гладкими стосовно факторної бази, або *B*-гладкими.

Як факторну базу B беремо множину простих чисел, яка складається з простих чисел *p*, які не перевищують задану межу L^a (вибирається з урахувань оптимальності). Межа L^a — це ще одне евристичне місце алгоритму.

Алгоритм працює в два етапи: етап збирання даних, які можуть привести до рівності квадратів, та етап оброблення даних, де він розміщує всю зібрану інформацію у матрицю і оброблює її для отримання розв'язку. Другий етап потребує великої ємності пам'яті і його важко розпаралелити.

Швидкість та результати роботи алгоритму залежать від розміру факторної бази та розміру інтервалу просіювання.

Якщо кількість простих чисел у факторній базі (розмір факторної бази) дуже мала, то розмір вектора степенів буде теж малим, що значно зменшує кількість операцій. Проблема полягає в тому, щоб знайти такі *B*-гладкі числа, які б входили в цю факторну базу. Чим менша факторна база, тим суттєво меншою є кількість *B*-гладких чисел, тобто необхідно значно збільшува-

ти інтервал просіювання. Якщо створити велику за розміром факторну базу, то постала б проблема розв'язання системи лінійних алгебричних рівнянь спеціального вигляду з матрицею великої розмірності, що потребує великої ємності пам'яті та ресурсів. Оптимальне значення розміру факторної бази пропонується у праці [3], яке обчислюється за формулою

$$A = L^a = \left(e^{\sqrt{\ln(n)\ln\ln(n)}} \right)^{\sqrt{2}/4} = L(n)^{\sqrt{2}/4} = L^{\sqrt{2}/4}. \qquad (1)$$

Ця формула не дає остаточної відповіді. Для кожного випадку найкращий розмір факторної бази є індивідуальним і може відрізнятися від значення, отриманого за формулою (1).

Наприклад, коли факторизували RSA-129 у 1994 р., використовували факторну базу простих чисел розміром 534339.

Інтервал просіювання повинен бути таким, щоб кількість *B*-гладких була більшою за кількість елементів у кожному векторі. Але цієї умови не достатньо. Можна скласти матрицю, де кількість векторів більша за кількість елементів у кожному векторі, та отримати хибний розв'язок. У такому випадку знадобиться розширити інтервал просіювання для отримання додаткових векторів. Для загального випадку (згідно з працею [3]) визначити розмір інтервалу просіювання можна за формулою

$$M_{\max} = L^b = \left(e^{\sqrt{\ln(n)\ln\ln(n)}} \right)^{3\sqrt{2}/4} = L(n)^{3\sqrt{2}/4} = L^{3\sqrt{2}/4}. \qquad (2)$$

Якщо після ділення числа *M* на всі прості числа з факторної бази *B*, залишок не дорівнює одиниці, то таке число відкидаємо. Додатковий аналіз цих чисел може надати більшу кількість векторів для побудови матриці.

Основною проблемою методу квадратичного решета є пошук достатньої кількості *B*-гладких чисел. Тому пошук способів отримання додаткових варіантів залишків, що можуть розглядатися як *B*-гладкі числа, є актуальним завданням.

Додатковий аналіз *B*-гладких чисел згадується в літературі [2, 5, 6]. Пропонується запам'ятовувати *B*-гладкі з неодиничним простим залишком. Для знаходження *B*-гладких з однаковими залишками потрібно використовувати їх разом.

Пропонується розглянути неодиничні залишки, які є квадратами простих чисел. Вектори таких чисел можна додавати до матриці, не враховуючи ці залишки. Як квадрати вони жодним чином не впливають на розв'язок. Якщо $y(a) = 7 \cdot 11^2 \cdot 23 \cdot 137^2$ та $y(b) = 7 \cdot 23$, тоді $y(a)y(b) = 7^2 \cdot 11^2 \cdot 23^2 \cdot 137^2$. За обраного максимального числа для факторної бази 23, вектор $y(a)$ увійде до матриці. Під час розв'язання матриці можна не враховувати 137^2, оскільки 137 має парний степінь. Такі залишки і називаються **умовно *B*-гладкими.**

Слід зазначити, що натепер в асиметричних криптосистемах використовуються числа розміру 1024 біт і більше. У цьому дослідженні використовувались числа розміром до 33 біт. Але результати дослідження дають основу для подальшого вивчення цього питання на числах розміру 1024 біт і більше.

ЗАСТОСУВАННЯ АНАЛІЗУ УМОВНО *B*-ГЛАДКИХ ЧИСЕЛ

Розглянемо на прикладі ефективність запропонованої модифікації. Оберемо $p = 401$ і $q = 103$. Ці прості числа створюють число для факторизації $pq = N = 41303$. Обчислимо за формулою (1) розмір факторної бази $A = 6$. За допомогою формули (2) отримаємо інтервал просіювання $M = 203$.

Після просіювання варіантів *y(x)* через факторну базу отримуємо *B*-гладкі числа. Ці числа наведено в табл. 1.

Таблиця 1. Результати просіювання варіантів $y(x)$

Знак числа	**Вектори степенів *B*-гладких чисел**						***B*-гладкі**
	2	11	19	23	29	37	
1	1	1	0	0	2	0	–18502
1	0	1	0	0	0	2	–15059
1	1	0	0	1	0	1	–1702
0	1	0	2	0	0	0	722
0	0	0	0	2	1	0	15341
0	1	1	0	1	0	1	18722

Цих чисел не достатньо для факторизації обраного *N*. Знайдемо умовно *B*-гладкі числа (табл. 2).

Таблиця 2. Результати додаткового просіювання варіантів $y(x)$

Знак числа	**Вектори степенів умовно *B*-гладких чисел**						**Дільники, які не входять до факторної бази**	**Умовно *B*-гладкі**
	2	11	19	23	29	37		
0	0	0	0	0	0	0	149^2	22201
0	1	0	0	0	0	0	131^2	34322
0	1	0	0	0	0	0	157^2	49298

Число 22201 не ввійшло до матриці, оскільки воно має прості дільники, які не потрапили у факторну базу. Число 22201 є квадратом, за допомогою якого отримуємо розв'язок. Числа 32322 та 49298 не є квадратами, але разом дають ще один розв'язок.

Розглянемо інший приклад. Оберемо $p = 11$ і $q = 601$, дістаємо $pq = N = 6611$. Обчислимо розмір факторної бази та інтервал просіювання $A = 5$, $M = 102$.

Після просіювання варіантів $y(x)$ через факторну базу отримуємо *B*-гладкі числа. Ці числа наведено в табл. 3.

Обчислюючи матрицю створену з векторів з табл. 3, дістаємо тільки хибні розв'язки. Знайдемо умовно *B*-гладкі числа (табл. 4).

Таблиця 3. Результати просіювання варіантів $y(x)$

Знак числа	Вектори степенів *B*-гладких чисел					*B*-гладкі
	2	5	17	29	31	
1	1	1	1	1	0	–4930
1	0	1	0	1	1	–4495
1	1	1	2	0	0	–2890
1	1	0	1	1	0	–986
1	0	0	1	0	1	–527
1	1	2	0	0	0	–50
0	0	1	0	2	0	4205
0	1	1	1	0	1	5270

Таблиця 4. Результати додаткового просіювання варіантів $y(x)$

Знак числа	Вектори степенів умовно *B*-гладких чисел					Дільники, які не входять до факторної бази	Умовно *B*-гладкі
	2	**5**	**17**	**29**	**31**		
1	1	0	0	0	0	41^2	–3362
0	0	1	0	0	0	37^2	6845

Число –3362 дозволило сформувати розв'язок з чисел: –4930, –4495, –3362 та –527.

Приклади випадків, де умовно *B*-гладкі числа входять до розв'язку, наведено в табл. 5.

Таблиця 5. Приклади факторизації з умовно *B*-гладкими числами

p	*q*	*N*	*B*-гладкі, які утворюють квадрат	Умовно *B*-гладкі	Множники умовно *B*-гладких чисел
27743	41203	1143094829	45292900	45292900	$5^2 \cdot 7^2 \cdot 673^2$
89	46411	4130579	-1496450, -5618	-1496450	$-1 \cdot 2 \cdot 5^2 \cdot 173^2$, $-1 \cdot 2 \cdot 53^2$
5647	40577	229138319	-29848630, -2996875, 11514850	-29848630,	$-1 \cdot 2 \cdot 5 \cdot 7653^2$, $-1 \cdot 5^2 \cdot 7 \cdot 137$, $2 \cdot 5^2 \cdot 41^2 \cdot 137$
29741	40087	1192227467	26759929	26759929	$7^2 \cdot 739^2$
30271	48533	1469142443	83375161	83375161	$23^2 \cdot 739^2$
30707	32089	985356923	477481	477481	691^2
31729	32423	1028749367	120409	120409	347^2
32443	45137	1464379691	40284409	40284409	$11^2 \cdot 577^2$
32887	39371	1294794077	10510564	10510564	$2^2 \cdot 1621^2$
6163	44777	275960651	-22386875, -2107, 23952473	23952473	$-1 \cdot 5^4 \cdot 7^2 \cdot 17 \cdot 43$, $-1 \cdot 7^2 \cdot 43$, $17 \cdot 1187^2$
36353	39511	1436343383	2493241	2493241	1579^2
37561	43067	1617639587	7579009	7579009	2753^2

Продовження табл. 5

38239	45413	1736547707	12866569	12866569	$17^2 \cdot 211^2$
39157	45119	1766724683	8886361	8886361	$11^2 \cdot 271^2$
40577	46811	1899449947	9715689	9715689	$3^2 \cdot 1039^2$
41719	45137	1883070503	2920681	2920681	1709^2
6359	43051	273761309	-38568413 -23710340 -6177145 -685684	-38568413 -23710340	$-1 \cdot 13 \cdot 41 \cdot 269^2$, $-1 \cdot 2^2 \cdot 5 \cdot 37 \cdot 179^2$, $-1 \cdot 5 \cdot 13 \cdot 29^2 \cdot 113$, $-1 \cdot 2^2 \cdot 37 \cdot 41 \cdot 113$
44867	47911	2149622837	2316484	2316484	$2^2 \cdot 761^2$
45403	46589	2115280367	351649	351649	593^2
48193	48539	2339240027	29929	29929	173^2

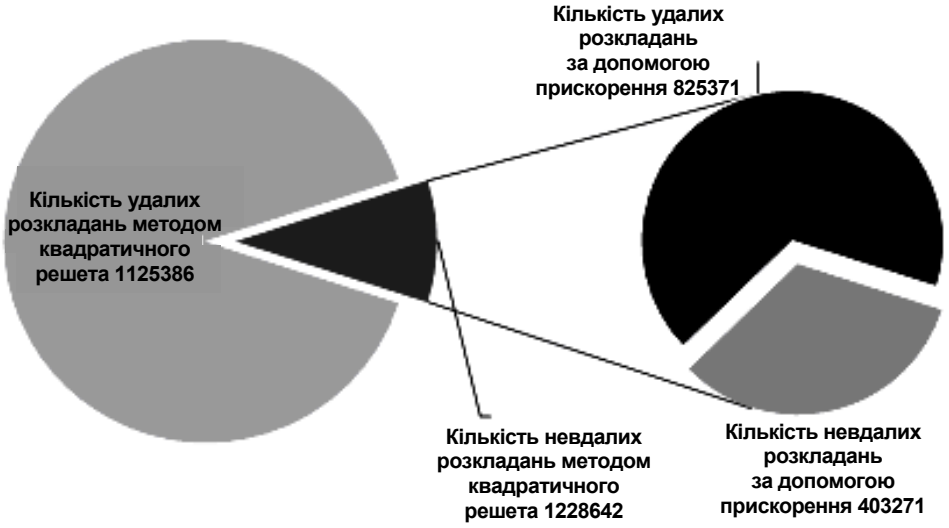
ПОРІВНЮВАЛЬНА ОЦІНКА АНАЛІЗУ УМОВНО *B*-ГЛАДКИХ ЧИСЕЛ

Додаткові вектори у сформованій матриці дозволили отримати розв'язок без розширення факторної бази або інтервалу просіювання.

Отримати числа, у яких залишок є квадратом, можна доволі часто. За першими 5000 простими числами сформовано $12.5 \cdot 10^6$ можливих варіантів *N*, знайдено додаткові вектори у 99% випадках.

Корисну дію цього методу можна побачити, якщо обрати випадки, у яких базовий алгоритм квадратичного решета за рекомендованими у працях [2, 4, 6] розмірами факторної бази та інтервалу просіювання, обчислених за формулами (1) і (2), не зміг знайти розв'язок, і застосувати аналіз $y(x)$, у яких залишок після просіювання є простим числом у парному степені.

У 7% випадках модифікований алгоритм зміг факторизувати число (див. рисунок).



Діаграма ефективності аналізу умовно *B*-гладких чисел

Варто зазначити, що якщо для порівнювального аналізу взяти меншу кількість простих чисел, починаючи не з першого простого числа, отримаємо кращі результати. Наприклад, якщо взяти тисячу простих чисел, починаючи з простого числа з порядковим номером 4000 або 5000, знайдемо, що модифікований алгоритм зміг факторизувати всі числа.

ОЦІНЮВАННЯ СКЛАДНОСТІ ТА ЧАСУ ВИКОНАННЯ

Складання матриці для стандартного алгоритму квадратичного решета потребує кількості елементів L^{2a} [3]. Кількість потрібних варіантів $y(x)$ (*B*-гладких) для стандартного квадратичного решета можна розрахувати за формулою L^{2a+1}.

Усі залишки *y(x)* уже отримано, їх пошук не потрібен. Для застосування аналізу умовно *B*-гладких чисел необхідно для кожного варіанта $y(x)$ запам'ятовувати залишок (якщо він є квадратом), тому потрібна ємність пам'яті збільшується і дорівнює L^{2a+1}.

На перший погляд додатковий аналіз варіантів $y(x)$ робить алгоритм складнішим і збільшує час його роботи — додаткове обчислення квадратного кореня з усіх залишків $y(x)$, більших за одиницю. Однак із застосуванням додаткового аналізу варіантів $y(x)$ кількість придатних *y(x)* збільшується (за рахунок умовно *B*-гладких) на деяке γ і становить $b = a + (4a)^{-1} + \gamma$. Значення b — кількість ітерацій в алгоритмі — обирається таким, щоб кількість придатних варіантів $y(x)$ становила L^a, тому $b = a + (4a)^{-1} - \gamma$. Як бачимо ця кількість зменшилась.

Беручи до уваги те, що b — показник степені інтервалу просіювання L^b, тоді кожне знайдене умовно *B*-гладке значення значно зменшує інтервал просіювання.

Оцінити швидкість модифікованого алгоритму можна за формулою

$$L^{\max\{2a+1,\,a+(4a)^{-1}-y,3a\}}.$$

Швидкість просіювання зменшилась на γ, де γ — кількість елементів $y(x)$ доданих умовно *B*-гладких залишків.

ВИСНОВКИ

Швидкість роботи методу квадратичного решета залежить від таких евристичних значень, як розмір факторної бази та інтервал просіювання. На основі проведених численних експериментів показано, що використання умовно *B*-гладких чисел дозволяє факторизувати число у тих випадках, коли базовий алгоритм квадратичного решета (за стандартного інтервалу просіювання та розміру факторної бази) не зміг сформувати матрицю для отримання розв'язку.

Модифікований алгоритм зміг зменшити кількість невдалих факторизацій з 11% до 3% відносно базового алгоритму квадратичного решета.

Швидкість просіювання модифікованого алгоритму зменшилась на γ. Для кожного випадку значення γ є різним і дорівнює кількості елементів $y(x)$ доданих завдяки використанню умовно *B*-гладких чисел.

Отримані результати є підставою для подальшого дослідження на числах 1024 біт і більше.

ЛІТЕРАТУРА

1. *Pomerance C.* The quadratic sieve factoring algorithm / C. Pomerance // Advances in Cryptology (T. Beth, N. Cot and I. Ingemarrson eds.), Lecture Notes in Comput. Sei. — Paris, 1985. — P. 169–182.
2. *Lindquist E.* The Quadratic Sieve Factoring Algorithm / E. Lindquist // Math 488: Cryptographic Algorithms, Diciembre. — New York, 2001. — P. 1–11.
3. *Pomerance C.* Analysis and comparison of some integer factoring algorithms / C. Pomerance // Mathematisch Centrum Computational Methods in Number Theory, Pt. 1. — Amsterdam: Math Centre Tract 154, 1982. — P. 89–139.
4. *Pomerance C.* Smooth numbers and the quadratic sieve / C. Pomerance // Proc. of an MSRI workshop. — New York: Proc. Amer. Math. Soc. 115, 2008. — P. 69–81.
5. *Song Y.* Quadratic Sieve / Y. Song // Primality Testing and Integer Factorization in Public-Key Cryptography Second Edition. — New York: Springer, 2008. — P. 234–239.
6. *Crandall R.* Smooth numbers and the quadratic sieve / R. Crandall, C. Pomerance // Prime Numbers A Computational Perspective Second Edition. — New York: Springer, 2005. — P. 261–315.
7. *Горбенко И.Д.* Анализ каналов уязвимости системы RSA / И.Д. Горбенко, В.И. Долгов, А.В. Потий, В.Н. Федорченко // Безопасность информации. — 1995. — № 2. — С.22–26.
8. *Brown D.* Breaking RSA May Be As Difficult As Factoring [Електронний ресурс] / Daniel R.L. Brown // Cryptology ePrint Archive. — 2005. — Режим доступу: https://eprint.iacr.org/2005/380.

Надійшла 24.07.2017