COMPUTATIONAL METHODS

УДК 003.26:511.337

СТЕПАН ВИННИЧУК,
ВІТАЛІЙ МІСЬКО

УДОСКОНАЛЕННЯ МЕТОДУ КВАДРАТИЧНОГО РЕШЕТА НА ОСНОВІ ВИКОРИСТАННЯ РОЗШИРЕНОЇ ФАКТОРНОЇ БАЗИ ТА ФОРМУВАННЯ ДОСТАТНЬОЇ КІЛЬКОСТІ B - ГЛАДКИХ ЧИСЕЛ

В інформаційно-телекомунікаційних системах для рішення задачі захисту інформації часто використовують RSA алгоритм. В основі криптостійкості найбільш популярного сьогодні асиметричного криптоалгоритму RSA є складність факторизації великих цілих чисел. Метод квадратичного решета є найкращим відомим методом факторизації чисел, розміром менше 110 десяткових знаків. Найбільш затратною за часом частиною алгоритму квадратичного решета є процес просіювання. Розмір факторної бази – це один з ключових параметрів, що визначають ефективність алгоритму просіювання. Надто великий розмір факторної бази потребує пошуку великої кількості *B*-гладких чисел, що збільшує загальний час виконання алгоритму. Коли розмір менший за необхідний, не вдасться знайти достатню кількість *B*-гладких чисел. У даній статті запропоновано метод визначення та застосування достатнього розміру B-гладких чисел при збільшенні розміру факторної бази вдвічі в порівнянні з базовим алгоритмом квадратичного решета. При розширенні факторної бази збільшується кількість чисел *N*, які можна розкласти на множники методом квадратичного решета. Відмічається також, що її збільшення призводить до росту обчислювальної складності, оскільки доцільно знаходити більшу кількість *B*-гладких чисел. Проте при проведенні чисельних експериментів, де розмір факторної бази збільшувався двічі, виявилося, що з використанням запропонованого алгоритму час, необхідний для пошуку достатньої кількості *B*-гладких чисел, навпаки зменшувався. За результатами обчислень встановлено, що при використанні вдвічі розширеної факторної бази час, необхідний для формування достатньої кількості елементів факторної бази зменшився на 29%, загальна кількість пробних значень, що досліджуються, із інтервалу просіювання при знаходженні *B*-гладких чисел зменшилася в 2,85 рази.

Ключові слова: квадратичне решето, розширена факторна база, достатня кількість *B*-гладких, інтервал просіювання, прості.

Вступ. Протягом багатьох років алгоритм RSA активно використовується як у вигляді самостійних криптографічних продуктів, так і в якості вбудованих засобів в популярних додатках. Відкрите шифрування на базі алгоритму RSA застосовується в популярному пакеті шифрування PGP, операційній системі Windows, різних Інтернет-браузерах, банківських комп'ютерних системах. Крім того, різні міжнародні стандарти шифрування з відкритим ключем і формування цифрового підпису використовують RSA в якості основного алгоритму. Найбільш поширена атака на цей криптоалгоритм заснована на факторизації публічного ключа [7], [8]. Якщо факторизація успішна, то всі повідомлення зашифровані відкритим ключем можуть читатися.

Метод квадратичного решета (Quadratic sieve algorithm, QS) відноситься до найшвидших алгоритмів факторизації [1]. Поступається методу решета числового поля. Проте для чисел розміром до 110 десяткових знаків і досі є найкращим. Зменшення обчислювальної складності методу квадратичного решета, дасть змогу покращити процес криптоаналізу алгоритму RSA. Тому дослідження нових способів зменшення його обчислювальної складності є актуальним.

Аналіз останніх досліджень і публікацій. В методі квадратичного решета шукають числа, квадрати яких за модулем *N* (число, яке факторизується) співпадають. Це здебільшого приводить до факторизації *N*.

Алгоритм методу QS працює в два етапи: етап збору даних, де збирається інформація, що може привести до рівності квадратів за модулем *N*, та етап обробки даних, де він розміщує всю зібрану інформацію у матрицю та оброблює її для отримання рішення [2].

На першому етапі вибирається інтервал просіювання, будується факторна база та реалізується процедура просіювання.

Для загального випадку (згідно з [3]), отримати інтервал просіювання можна за формулою:

$$L^b = \left(e^{\sqrt{ln(N)lnln(N)}}\right)^{3\sqrt{2}/4} = L(N)^{3\sqrt{2}/4} = L^{3\sqrt{2}/4} \quad (1)$$

Розмір факторної бази згідно [2, 4, 6] рекомендується вибирати за формулою

$$L^a = \left(e^{\sqrt{\ln(N)\ln\ln(N)}}\right)^{\sqrt{2}/4} = L(N)^{\sqrt{2}/4} = L^{\sqrt{2}/4}. \quad (2)$$

Найбільш затратною за часом частиною алгоритму квадратичного решета є процес просіювання. Під час просіювання шукають пари чисел (A,B), які задовольняють умові

$$B = A^2 (mod\ N), \quad (3)$$

де число *B* розкладається на множники, що є елементами факторної бази. Такі числа *B* називають *B*-гладкими.

Розмір факторної бази – це один з ключових параметрів, що визначають ефективність алгоритму просіювання. Надто великий розмір факторної бази потребує пошуку великої кількості *B*-гладких чисел, що збільшує загальний час виконання алгоритму [2], [5], [6]. Коли розмір менший за необхідний, не вдасться знайти достатню кількість *B*-гладких чисел. Співвідношення (2) – це рекомендована величина числа елементів факторної бази, отримана на основі чисельних експериментів. Тоді алгоритм методу QS шукає *B*-гладкі числа у кількості не менше ніж $L^a + 2$. Якщо ж достатньої кількості *B*-гладких чисел не знайдено на інтервалі просіювання, можна збільшувати як розмір факторної бази, так і інтервал просіювання, що призводить до суттєвого росту часу виконання алгоритму.

Ідея досліджень, результати яких подаються в даній статті, полягає у використанні початкового розміру факторної бази $L_{\max} > L^a$ та визначенні достатнього розміру L^*, який може виявитися меншим за L^a.

Постановка задачі. Для досягнення поставленої мети вирішувалися наступні задачі:

1. Розробити алгоритм визначення мінімальної кількості *B*-гладких чисел.
2. Для етапу просіювання, провести порівняльний аналіз методу визначення мінімальної кількості *B*-гладких чисел зі стандартним алгоритмом.

Виклад основного матеріалу дослідження.

Метод вибору мінімальної достатньої кількості *B*-гладких чисел. У рамках цього дослідження вважатимемо вирішеною задачу пошуку розмірів інтервалу просіювання та факторної бази з L_{max} елементів.

В алгоритмі мінімальної достатньої кількості *B*-гладких чисел (Method of least *B*-smooth, MLB**)**, що пропонується, реалізується вибір достатньої кількості *B*-гладких чисел при розмірі факторної бази $L_{\max} > L^a$, використовуються два додаткові вектори $Ve[L_{\max}+1]$ та $Vf[L_{\max}+1]$. Кожен їх елемент поставлений у відповідність елементу факторної бази. В цих векторах нульовій клітинці відповідає знак *B*-гладкого числа, а довільній іншій клітинці – відповідний порядковий номер елемента факторної бази, де елементи факторної бази розміщені в порядку зростання їх значень.

Вектор $Ve[L_{\max}+1]$ – це інформація про показники степенів отриманого нового *B*-гладкого числа, де в нульовій клітинці значенню 1 відповідає від'ємне значення *B*-гладкого числа, а значенню 0 – додатне. В довільній іншій клітинці *k* вектора *Ve* вказано показник степеня елемента факторної бази за номером *k*, що є дільником *B*-гладкого числа, а інакше нуль.

В кожній клітинці k вектора $Vf[L_{max}+1]$ вказано кількість B-гладких чисел, для яких порядковий номер s максимального за значенням елемента факторної бази, що є дільником B-гладкого числа з непарним показником степеня, не перевищує k ($s \le k$).

В алгоритмі MLB, що пропонується, використовуватиметься також вектор $VB[L_{max}+2]$ – в кожній його клітинці t міститься інформація про значення числа з інтервалу просіювання $(-L^b, L^b)$ на основі якого отримано B-гладке число з номером t, а також вектор $VM[L_{max}+2]$. Клітинці t вектора VM відповідає B-гладке число з номером t, для якого задається порядковий номер s максимального за значенням елемента факторної бази, що є дільником цього B-гладкого числа з непарним показником степеня.

Визначення початкових значень елементів векторів Vf, VM та VB, їх зміни при отриманні нового B-гладкого числа та умови достатності кількості B-гладких чисел представлені такими кроками алгоритму MLB:

1. Присвоїти довільному k-у елементу вектора Vf значення $k + 2$, довільному з елементів векторів Ve та VB значення 0. Лічильнику nb B-гладких чисел присвоїти значення 0.
2. На етапі проріджування при отриманні B-гладкого числа B з номером nb, на основі числа x з інтервалу просіювання, сформувати вектор Ve показників степенів для дільників B та визначити найбільший порядковий номер s ненульового непарного елемента в ньому. Присвоїти $nb = nb + 1$; $VB[nb] = x$; $VM[nb] = s$.
3. Для всіх елементів вектора Vf, починаючи з номера s, зменшити їх значення на одиницю.
4. Кроки 2 та 3 продовжувати до тих пір, поки для одного з елементів вектора Vf , наприклад k, не буде виконана умова $Vf[k] = 0$, або для додатного B всі показники степенів у векторі Ve парні. Якщо для додатного B всі показники степенів у векторі Ve парні, перейти до кроку 5, а при $Vf[k] = 0$ – до кроку 6.
5. Отримуємо множники N згідно з методом факторизації Ферма і завершити роботу алгоритму.
6. Прийняти $L^* = k$. Сформувати матрицю M, що відповідає $k + 2$ - м B-гладким числам, для кожного з яких з порядковим номером t $VM[t] \le k$. Для формування j-го рядка матриці M необхідно:
   a. знайти t_j, для якого $VM[t] \le k$;
   b. знайти значення B-гладкого числа за формулою $B = x^2 - N$, де $x = VM[t]$;
   c. сформувати вектор Ve показників степенів елементів факторної бази, дільників B та для непарних їх значень у відповідному стовпчику матриці M записати 1, а в інших випадках 0.
7. Опрацювати матрицю та з'ясувати чи отримане значення кореня не дорівнює N. Якщо ні, то задача факторизації вирішена, а інакше перейти до кроку 7.
8. Видалити інформацію про B-гладке число, що відповідає рядку $j0$ з нульовими значеннями елементів перетвореної матриці. Присвоїти:
   a. $Vf[i] = Vf[i] + 1$, де $i \ge VM[t_{j0}]$;
   b. $VM[t_{j0}] = VM[nb]$; $VM[nb] = 0$;
   c. $Vb[t_{j0}] = Vb[nb]$; $Vb[nb] = 0$;
   d. $nb = nb - 1$.
   e. Перейти до кроку 2.

Приклади застосування алгоритму MLB. Приклад 1. Нехай $p = 23$, $q = 97$, $N = 2231$, розмір факторної бази: $L^a = 4$, інтервал просіювання: $L^b = 67$, початкове значення $X0 = xx = \sqrt{N} = 48$.

Елементи факторної бази: 2, 5, 11, 17.

Елементи факторної бази подвоєної довжини: 2, 5, 11, 17, 37, 43, 59, 71.

Дані про B-гладкі числа, їх множники та значення елементів векторів $Ve[]$ та $Vf[]$ за елементами бази наведено в табл. 1.

Таблиця 1 – Дані про *B*-гладкі числа для $N = 2231$

Номер *B*-гладкого	*dx*	$R = \sqrt{X^2 - N}$	Вектори *Ve* та *Vf*	Знак *R*	Елементи факторної бази							
					2	5	11	17	37	43	59	71
0	–	–	*Ve*	0	0	0	0	0	0	0	0	0
			Vf	2	3	4	5	6	7	8	9	10
1	1	170	*Ve*	0	1	1	0	1	0	0	0	0
			Vf	2	3	4	5	5	6	7	8	9
2	3	370	*Ve*	0	1	1	0	0	1	0	0	0
			Vf	2	3	4	5	5	5	6	7	8
3	4	473	*Ve*	0	0	0	1	0	0	1	0	0
			Ve	2	3	4	5	5	5	5	6	7
4	5	578	*Ve*	0	1	0	0	2	0	0	0	0
			Vf	2	2	3	4	4	4	4	5	6
5	11	1250	*Ve*	0	1	4	0	0	0	0	0	0
			Vf	2	1	2	3	3	3	3	4	5
6	12	1369	*Ve*	**0**	**0**	**0**	**0**	**0**	**2**	**0**	**0**	**0**
			Vf									

Для останнього з *B*-гладких чисел отримано вектор його множників *Ve*, в якому всі показники степенів парні, що відповідає умовам кроку 5 алгоритму MLB.

Отримані множники *N:* $p = X - Y = 60 - 37 = 23$, $p = X + Y = 60 + 37 = 97$.

Приклад 2. $p = 277$, $q = 2917$, $N = 808009$,

$L^a = 8$, $L^b = 555$, $X0 = xx = \sqrt{N} = 899$ю

Елементи факторної бази: 2, 3, 5, 11, 17, 31, 37, 47.

Елементи факторної бази подвоєної довжини: 2, 3, 5, 11, 17, 31, 37, 47, 53, 59, 61, 67, 71, 79, 83, 89.

Дані про *B*-гладкі числа, їх множники та значення елементів векторів *Ve*[] та *Vf*[] за елементами бази наведено в табл. 2.

Таблиця 2 – Дані про *B*-гладкі числа для $N = 808009$

№	dx	R	*Ve*, *Vf*	Знак *R*	Елементи факторної бази														
					2	5	11	17	37	43	59	71	2	5	11	17	37	43	59	71
0	–	–	*Ve*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
			Vf	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0	192	*Ve*	**0**	**6**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
			Vf	2	3	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2	2	3792	*Ve*	0	6	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0
			Vf	2	3	3	4	5	6	7	8	9	10	11	12	13	14	14	15	16
3	-2	-3400	*Ve*	**1**	**3**	**0**	**2**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
			Ve	2	3	3	4	5	5	6	7	8	9	10	11	12	13	13	14	15
4	4	7400	*Ve*	0	3	0	2	0	0	0	1	0	0	0	0	0	0	0	0	0
			Ve	2	3	3	4	5	5	6	6	7	8	9	10	11	12	12	13	14
5	5	9207	*Ve*	0	0	3	0	1	0	1	0	0	0	0	0	0	0	0	0	0
			Vf	2	3	3	4	5	5	5	5	6	7	8	9	10	11	11	12	13
6	6	11016	*Ve*	**0**	**3**	**4**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
			Vf	2	3	3	4	5	4	4	4	5	6	7	8	9	10	10	11	12
7	-6	-10560	*Ve*	**1**	**6**	**1**	**1**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
			Vf	2	3	3	4	4	3	3	3	4	5	6	7	8	9	9	10	11
8	8	14640	*Ve*	0	4	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0
			Vf	2	3	3	4	4	3	3	3	4	5	6	6	7	8	8	9	10

Продовження таблиці 1

9	-10	-17688	*Ve*	1	3	1	0	1	0	0	0	0	0	0	0	1	0	0	0
			Vf	2	3	3	4	4	3	3	3	4	5	6	6	6	7	7	8	9
10	12	21912	*Ve*	0	3	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0
			Vf	2	3	3	4	4	3	3	3	4	5	6	6	6	7	7	7	8
11	-12	-21240	*Ve*	1	3	2	1	0	0	0	0	0	0	1	0	0	0	0	0	0
			Vf	2	3	3	4	4	3	3	3	4	5	5	5	5	6	6	6	7
12	14	25560	*Ve*	0	3	2	1	0	0	0	0	0	0	0	0	0	1	0	0	0
			Vf	2	3	3	4	4	3	3	3	4	5	5	5	5	5	5	5	6
13	16	29216	*Ve*	0	5	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0
			Vf	2	3	3	4	4	3	3	3	4	5	5	5	5	5	5	4	5
14	-16	-28320	*Ve*	1	5	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0
			Vf	2	3	3	4	4	3	3	3	4	5	4	4	4	4	4	3	4
15	-21	-37125	*Ve*	**1**	**0**	**3**	**3**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
			Vf	2	3	3	4	3	2	2	2	3	4	3	3	3	3	3	2	3
16	-22	-38880	*Ve*	**1**	**5**	**5**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
			Vf	2	3	3	3	2	1	1	1	2	3	2	2	2	2	2	1	2
17	23	42075	*Ve*	**0**	**0**	**2**	**2**	**1**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
			Vf	2	3	3	3	2	0	0	0	1	2	1	1	1	1	1	0	1

У результаті знайдено 17 *B*-гладких чисел, серед яких для 7 (в табл. 2 степені дільників для них виділені напівжирним) максимальний за значенням дільник не перевищує 5-го елемента факторної бази, рівного 37. Такі *B*-гладкі числа формують матрицю *M*, що містить 6 стовпчиків та 7 рядків, де 4 та 5 рядки співпадають:

Таблиця 3 – Матриця *M*

Знак числа	2	5	11	17	37	*B* – гладкі
0	**0**	**1**	**0**	**0**	**0**	192
1	**1**	**0**	**0**	**0**	**1**	-3400
0	**1**	**0**	**0**	**0**	**1**	11016
1	**0**	**1**	**1**	**1**	**0**	-10560
1	**0**	**1**	**1**	**1**	**0**	-37125
1	**1**	**1**	**1**	**0**	**0**	-38880
0	**0**	**0**	**0**	**1**	**1**	42075

Приклад 3. p =509, q = 2857, N = 1454213, L^a = 8, L^b = 669, $X0$ = xx = $\sqrt{N}$ = 1206.

Елементи факторної бази: 2, 17, 41, 53, 61, 89, 97, 103.

Елементи факторної бази подвоєної довжини (в дужках вказано порядковий номер елемента):

2 (1), 17 (2), 41 (3), 53 (4), 61 (5), 89 (6), 97 (7), 103 (8), 107 (9), 113 (10), 131 (11), 163 (12), 167 (13), 173 (14), 179 (15), 181 (16), 191 (17).

При однаковому інтервалі просіювання у випадку факторної бази, що містить 8 елементів, не знайдено достатньої кількості *B*-гладких чисел. При використанні розширеної факторної бази отримано 18 *B*-гладких чисел, серед яких виявилося чотири, у яких максимальне значення дільника з непарним показником степеня не перевищує 17. Дані про такі *B*-гладкі числа, їх множники та значення елементів векторів *Ve* та *Vf* за елементами бази наведено в табл. 4.Виділені *B*-гладкі числа формують матрицю *M* (див. табл. 5), що містить 3 стовпчики та 4 рядки, де 2 та 3 рядки співпадають.

При розширенні факторної бази збільшується кількість чисел *N*, які можна розкласти на множники методом квадратичного решета. Відмічається також, що її збільшення призводить

до росту обчислювальної складності, оскільки доцільно знаходити більшу кількість *B*-гладких чисел. Проте при проведенні чисельних експериментів, де розмір факторної бази збільшувався двічі, виявилося, що з використанням алгоритму MLB час, необхідний для пошуку достатньої кількості *B*-гладких чисел, навпаки зменшувався.

Таблиця 4 – Дані про *B*-гладкі числа для $N = 1454213$

№	dx	R	*Ve*, *Vf*	Знак *R*	Порядкові номери елементів факторної бази															
					1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	–	–	*Ve*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
			Vf	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
2	-4	-9409	*Ve*	1	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0
			Vf	1	2	3	4	5	6	7	8	8	9	10	11	12	13	14	15	16	17
4	-12	-28577	*Ve*	1	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
			Vf	1	2	2	3	4	5	6	7	7	8	9	10	11	12	13	13	14	15
16	-216	-474113	*Ve*	1	0	1	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0
			Ve	1	2	1	2	2	3	4	5	5	5	5	6	7	6	4	4	3	3
18	269	721412	*Ve*	**0**	**2**	**1**	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0
			Ve	1	2	0	1	1	2	3	4	4	4	4	5	6	5	2	2	1	1

Таблиця 5 – Матриця *M*

Знак числа	2	17	*B* – гладкі
1	**0**	**0**	-9409
1	**0**	**1**	-28577
1	**0**	**1**	-474113
0	**0**	**1**	721412

У чисельних експериментах досліджувалися 10^6 відносно малих чисел, що є добутками простих p в діапазоні $p = 277 \div 8369$ та простих q в діапазоні $q = 37811 \div 48589$. За результатами обчислень встановлено, що при використанні вдвічі розширеної факторної бази:

– час, необхідний для формування достатньої кількості елементів факторної бази зменшився на 29%;

– загальна кількість пробних значень, що досліджуються, із інтервалу просіювання при знаходженні *B*-гладких чисел (якщо достатню їх кількість не вдалося знайти, вважалося, що використано весь інтервал просіювання від $-L^b$ до L^b) зменшилася від 1056931741 до 370331821, тобто в 2,85 рази;

– кількість чисел *N*, для яких не вдалося знайти їх множники зменшилася із 39612 до 227, тобто на 3.96% від загальної кількості *N*. У випадку числа елементів факторної бази, рівного L^a, коли не використовувався спосіб вибору достатньої кількості *B*-гладких чисел, кількість чисел *N*, для яких не вдалося знайти їх множники, становила 40084;

– розмірність матриці *M* при збільшенні факторної бази вдвічі зростає в середньому в 1,8 рази.

Висновки. В результаті чисельних експериментів показано, що для найбільш затратного за часом етапу методу квадратичного решета, – пошуку *B*-гладких чисел, – у випадку збільшення факторної бази вдвічі та формування достатньої кількості *B*-гладких чисел:

– у середньому в 2,85 рази зменшилася кількість використаних пробних значень з інтервалу просіювання при збільшенні розміру матриці *M* в 1.8 рази;

– на 29% зменшився загальний час на просіювання пробних значень та формування достатньої кількості *B*-гладких для 10^6 варіантів факторизованих чисел *N*;

– на 3,96 відсотків від загальної кількості *N*, що розкладалися на множники, збільшилась кількість вдалих факторизацій порівняно з базовим методом квадратичного решета.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

[1] C. Pomerance, “The quadratic sieve factoring algorithm”, in *Proc. of EUROCRYPT 84. A Workshop on the Theory and Application of Cryptographic Techniques*, Paris, 1984. pp. 169-182. doi: 10.1007/3-540-39757-4_17.
[2] Landquist E. “The Quadratic Sieve Factoring Algorithm”. [Online]. Available: http://www.cs.virginia.edu/crab/QFS_Simple.pdf . Accessed on: Sept. 19, 2017.
[3] C. Pomerance “Analysis and comparison of some integer factoring algorithms”, in *Computational Methods in Number Theory, vol. 154*, Amsterdam, Netherlands: Math. Centre Amsterdam, 1982, pp. 89-139.
[4] C. Pomerance, “Smooth numbers and the quadratic sieve”, *Algorithmic Number Theory*, vol. 44, pp. 69-81, 2008.
[5] Y.Y.Song, *Primality testing and integer factorization in public-key cryptography*, New York, USA: Springer Publishing, 2009. doi: 10.1007/978-0-387-77268-4.
[6] R. Crandall, and C. Pomerance, *Prime Numbers. A Computational Perspective*, New York, USA: Springer Publishing, 2005.
[7] И.Д. Горбенко, В.И. Долгов, А.В. Потий, и В.Н. Федорченко, “Анализ каналов уязвимости системы RSA”, *Безопасность информации*, № 2, с. 22-26, 1995.
[8] Daniel R. L. Brown, “Breaking RSA may be as difficult as factoring”. [Online]. Available: https://eprint.iacr.org/2005/380.pdf. Accessed on: Sept. 19, 2017.

Стаття надійшла до редакції 26 вересня 2017 року.

REFERENCE

[1] C. Pomerance, “The quadratic sieve factoring algorithm”, in *Proc. of EUROCRYPT 84. A Workshop on the Theory and Application of Cryptographic Techniques*, Paris, 1984. pp. 169-182. doi: 10.1007/3-540-39757-4_17.
[2] Landquist E. “The Quadratic Sieve Factoring Algorithm”. [Online]. Available: http://www.cs.virginia.edu/crab/QFS_Simple.pdf . Accessed on: Sept. 19, 2017.
[3] C. Pomerance “Analysis and comparison of some integer factoring algorithms”, in *Computational Methods in Number Theory, vol. 154*, Amsterdam, Netherlands: Math. Centre Amsterdam, 1982, pp. 89-139.
[4] C. Pomerance, “Smooth numbers and the quadratic sieve”, *Algorithmic Number Theory*, vol. 44, pp. 69-81, 2008.
[5] Y.Y.Song, *Primality testing and integer factorization in public-key cryptography*, New York, USA: Springer Publishing, 2009. doi: 10.1007/978-0-387-77268-4.
[6] R. Crandall, and C. Pomerance, *Prime Numbers. A Computational Perspective*, New York, USA: Springer Publishing, 2005.
[7] I.D Gorbenko, V.I. Dolgov, A.V. Potiy, and V.N. Fedochenko, “Analysis of RSA system vulnerability channels”, *Information security*, no. 2, pp. 22-26, 1995.
[8] Daniel R. L. Brown, “Breaking RSA may be as difficult as factoring”. [Online]. Available: https://eprint.iacr.org/2005/380.pdf. Accessed on: Sept. 19, 2017.

СТЕПАН ВИННИЧУК,
ВИТАЛИЙ МИСЬКО

УСОВЕРШЕНСТВОВАНИЕ МЕТОДА КВАДРАТИЧНОГО РЕШЕТА НА ОСНОВЕ ИСПОЛЬЗОВАНИЯ РОЗШИРЕННОЙ ФАКТОРНОЙ БАЗЫ И ФОРМИРОВАНИЯ ДОСТАТОЧНОГО КОЛИЧЕСТВА B-ГЛАДКИХ ЧИСЕЛ

В информационно-телекоммуникационных системах для решения задачи защиты информации часто используют RSA алгоритм. В основе криптостойкости наиболее

популярного сегодня ассиметричного криптоалгоритма RSA лежит сложность факторизации больших целых чисел. Метод квадратичного решета является лучшим методом факторизации чисел, размером меньше 110 десятичных знаков. Наиболее затратной по времени частью алгоритма квадратичного решета является процесс просеивания. Размер факторной базы – это один из ключевых параметров, который определяет эффективность алгоритма просеивания. Большой размер факторной базы требует поиска большого количества *B*-гладких чисел, что увеличивает общее время выполнения алгоритма. Когда размер меньше необходимого, тогда не удастся найти достаточного количества *B*-гладких чисел. В данной статье предложено метод определения и использования достаточного размера *B*-гладких чисел, при увеличении размера факторной базы вдвое, в сравнении с базовым алгоритмом квадратичного решета. При расширении факторной базы увеличивается количество чисел *N,* которые можно разложить на множители методом квадратичного решета. Отмечается также, что её увеличение приводит к росту вычислительной сложности, поскольку целесообразно находить большее количество *B*-гладких чисел. Но, при проведении численных экспериментов, где размер факторной базы увеличивается вдвое, обнаружено, что с использованием предложенного алгоритма время, необходимое для поиска достаточного количества *B*-гладких чисел, наоборот уменьшается.

Ключевые слова: квадратичное решето, расширенная факторная база, достаточное количество B-гладких, интервал просеивания, простые.

STEPAN VYNNYCHUK,
VITALII MISKO

IMPROVEMENT OF THE QUADRATIC SIEVE METHOD ON THE BASIS OF THE EXTENDED FACTOR BASE AND USING AVAILABLE QUANTITY OF B – SMOOTH NUMBERS

In information and telecommunication systems, RSA algorithms are often used to solve information security problems. At the core of the cryptostability of the most popular today asymmetric cryptographic algorithm RSA is the complexity of the factorization of large integers. Quadratic sieve method is the best for factorization of integers under 110 decimal digits or so. The most time consuming part of the algorithm of a quadratic sieve is the sieving process. The size of the factor base is one of the key parameters that determine the effectiveness of the sieving algorithm. Too large factor base requires the search for a large number of *B*-smooth numbers, which increases the total execution time of the algorithm. When the size is less than necessary, it will not be possible to find a sufficient number of *B*-smooth numbers. In this paper, the method for determining and applying a sufficient size of *B*-smooth numbers with doubling the factor base size in comparison with the basic algorithm of a quadratic sieve is proposed. With the expansion of the factor base, the number of *N* numbers increases, which can be decomposed into factors by the quadratic sieve method. It is also noted that its increase leads to an increase in computational complexity, since it is advisable to find a greater number of *B*-smooth numbers. However, when conducting numerical experiments, where the size of the factor base increased twice, it turned out that using the proposed algorithm, the time necessary to find a sufficient number of *B*-smooth numbers, on the contrary, decreased.

Keywords: quadratic sieve, extended factor base, available quantity of B-smooth, sieving interval, prime.

Степан Дмитрович Винничук, доктор технічних наук, старший науковий співробітник, Інститут проблем моделювання в енергетиці імені Г. Є. Пухова Національної академії наук України; професор кафедри кібербезпеки та застосування автоматизованих інформаційних систем і технологій Національного технічного університету України “Київський політехнічний інститут імені Ігоря Сікорського”, Київ, Україна.

E-mail: vynnychuk@i.ua.

Віталій Миколайович Міська, аспірант, Інститут проблем моделювання в енергетиці імені Г. Є. Пухова Національної академії наук України, Київ, Україна.

E-mail: vitalii.misko@gmail.com.

Степан Дмитриевич Винничук, доктор технических наук, старший научный сотрудник, Институт проблем моделирования в энергетике им. Г.Е. Пухова Национальной академии наук Украины; профессор кафедры кибербезопасности и применения автоматизированных информационных систем и технологий, Институт специальной связи и защиты информации Национального технического университета Украины "Киевский политехнический институт имени Игоря Сикорского", Киев, Украина.

Виталий Николаевич Мисько, аспирант, Институт проблем моделирования в энергетике им. Г.Е. Пухова Национальной академии наук Украины, Киев, Украина.

Stepan Vynnychuk, doctor of technical science, senior researcher, Pukhov institute for modeling in energy engineering of National academy of sciences of Ukraine; professor at the cybersecurity and application of information systems and technologies academic department, Institute of special communication and information protection of National technical university of Ukraine "Igor Sikorsky Kyiv polytechnic institute", Kyiv, Ukraine.

Vitalii Misko, postgraduate student, Pukhov institute for modeling in energy engineering of National academy of sciences of Ukraine, Kyiv, Ukraine.