УДК 511 УДК 519.6

М.В. СЕМОТЮК
Інститут кібернетики ім. В.М. Глушкова НАН України, Київ, Україна,
e-mail: *semo@i.ua.*

ТЕОРЕТИКО-ЧИСЛОВІ МЕТОДИ ФАКТОРИЗАЦІЇ СКЛАДЕНИХ ЧИСЕЛ ТА ОБЧИСЛЕННЯ ДИСКРЕТНОГО ЛОГАРИФМА

Анотація. Стаття присвячена новому застосуванню теоретико-числових перетворень. Подання систем числення цими перетвореннями дає змогу створити принципово нові і ефективні алгоритми факторизації чисел, обчислення періоду показникової функції та дискретного логарифма. Алгоритм факторизації дозволяє за один прохід розкласти будь-який скінченний добуток на множники, він є точним тестом простоти чисел. Цей алгоритм ґрунтується на поданні систем числення теоретико-числовим перетворенням і не має аналогів, оскільки використовує тільки прості арифметичні дії. Властивості простоти чисел або інші властивості чисел не застосовуються. Отже, факторизація чисел, обчислення періоду показникової функції та дискретного логарифма є арифметичними операціями, що виконуються за скінченний час і належать до P-класу складності.

Ключові слова: множина, грані множини, алгебра, кільце лишків, модуль, аксіоматика цілих чисел, теоретико-числове перетворення, система числення, основа системи числення, факторизація, арифметична операція, період показникової функції, дискретний логарифм.

ВСТУП

Існує багато задач, для яких не знайдено поліноміального алгоритму, але не доведено, що його не існує, тому невідомо, чи належать такі задачі до класу складності P. Однією з таких задач є розкладання складеного числа на множники, яке коротко називають факторизацією. Факторизація великих чисел — надзвичайно трудомістке завдання навіть для сучасних комп'ютерів [1]. У роботі [2] показано, що існує метод факторизації складених чисел, який ґрунтується на дуалізмі (подвійності) операцій у кільці лишків за модулем і породжує систему рівнянь. Хоча метод дає змогу звести факторизацію великих чисел до факторизації малих чисел, він не дозволяє стверджувати, що факторизація чисел належить до класу P-складності, оскільки вимагає інших підходів до вирішення цієї задачі.

АКСІОМАТИКА ЦІЛОГО (АНТЬЄ)

Зафіксуємо число m і розглянемо множину

$$\mathbf{Z}_m = \{0, 1, 2, \ldots, m-1\} \tag{1}$$

всіх залишків від ділення елементів числової множини $\mathbf{Z}$ на m, які входять до множини $\mathbf{Z}_m$ лише один раз. Тоді стосовно множини (1) можна констатувати, що верхньою межею або супремумом цієї множини є вираз

$$\sup \mathbf{Z}_m = \min_{i=0}^{\infty} \left(m * \operatorname{int} \frac{z_i}{m} \right) = \min_{i=0}^{m-1} (m * \operatorname{int}_m (z_i)) \;\; \forall z_i \in \mathbf{Z} | z_i \neq 0, \tag{2}$$

де $\operatorname{int}_m()$ — ціла частина відносно модуля m (коротке позначення), а максимальний елемент цієї множини має вигляд

$$\max\{z_i\} = \max_{i=0}^{m-1} ((z_i) \bmod m). \tag{3}$$

Аналогічно нижня грань множини має вигляд

$$\inf \mathbf{Z}_m = \max_{i=0}^{m-1} ((z_i) \bmod 1) \quad \forall z_i \in \mathbf{Z}, \tag{4}$$

а мінімальний елемент дорівнює

$$\min\{z_i\} = \min_{i=0}^{m-1} (\mathrm{int}_1(z_i)). \tag{5}$$

Зауважимо, що в цій інтерпретації аксіоматики цілого

$$\sup \mathbf{Z}_m \neq \max\{z_i\} \quad \text{і} \quad \inf \mathbf{Z}_m \neq \min\{z_i\}.$$

Якщо множину цілих чисел задано інтервалом $k << p$, вважаючи, що $m = k / p$, $k > p > 0$ і k кратно p, то множину (1) можна отримати, застосовуючи грані

$$\sup \mathbf{Z}_m = \min_{i=0}^{m-1} \left(k / p \cdot \mathrm{int}\left(\frac{z_i}{k / p} \right) \right) = \min_{i=0}^{m-1} (k / p \cdot \mathrm{int}_{k/p}(z_i)) \quad \forall z_i \in \mathbf{Z}, \tag{6}$$

$$\inf \mathbf{Z}_m = \max_{i=0}^{m-1} ((z_i) \bmod p) \quad \forall z_i \in \mathbf{Z}, \tag{7}$$

а максимальний і мінімальний елементи мають вигляд

$$\max\{z_i\} = \max_{i=0}^{m-1} ((z_i) \bmod k / p) \quad \forall z_i \in \mathbf{Z}, \tag{8}$$

$$\min\{z_i\} = \min_{i=0}^{m-1} (\mathrm{int}_p(z_i)), \tag{9}$$

де $\mathrm{int}_p(z_i) = \mathrm{int}(z_i / p)$ — ціла частина стосовно p.

З аналізу виразів (6) і (7) випливає, що порядок обчислення граней множини (1) є залежним, тобто спочатку потрібно обчислити нижню межу, а потім верхню. Це також стосується виразів (8) і (9). Однак множина натуральних чисел входить у множину цілих чисел, яка належить до множини раціональних чисел, а множина раціональних чисел входить у множину дійсних чисел. Тоді фіксована нижня грань (4) поділяє множину раціональних чисел на множину цілих чисел і множину дробових чисел. Верхня грань (2) обмежує множину (1) таким чином, що в результаті маємо числову строго впорядковану множину, яку використовують сучасні комп'ютери. Ця верхня грань визначається довжиною розрядної сітки комп'ютера, а вихід за межі цієї грані викликає переповнення розрядної сітки і це є аварійним режимом. Разом з тим вважаючи, що модуль може бути і не цілим числом, адже не викликає сумніву обчислення аргументів тригонометричних функцій за модулем 2π, то вочевидь, що залишок або мантиса теж може бути меншим за одиницю. Тоді можна скористатися функцією з мов програмування fmod p:

$$(a) \,\mathrm{fmod}\, p = a - \mathrm{int}_p(a). \tag{10}$$

Це означає, що незалежно від того, чи є a і p цілими чи дробовими числами, а ділення a на p можливе тільки до остаточного значення частки, залишок може бути і меншим за одиницю. Отже, вирази (7) і (9) перепишемо

у вигляді

$$\inf \mathbf{Z}_m = \max_{i=0}^{m-1} ((z_i) \operatorname{fmod} p) \ \ \forall z_i \in \mathbf{R}, \tag{11}$$

$$\min\{z_i\} = \min_{i=0}^{m-1} ((z_i) \operatorname{fmod} p) \ \ \forall z_i \in \mathbf{R}. \tag{12}$$

Зауважимо, що множину (1) можна отримати застосуванням граней (6) і (11) до кожного елемента множини $\mathbf{R}$. Потужність отриманої цілочисельної множини, що розглядається як строго впорядкована множина, дорівнює різниці цих граней:

$$\operatorname{card} \mathbf{Z}_m = \tag{13}$$

$$= \sup \mathbf{Z}_m - \inf \mathbf{Z}_m = \min_{i=0}^{m-1} (k \ / \ p \cdot \operatorname{int}_{k/p}(z_i)) - \max_{i=0}^{m-1} ((z_i) \operatorname{fmod} p) \ \ \forall z_i \in \mathbf{R}.$$

Якщо елементи множини (1) отримано застосуванням задіяних граней, значення яких ми можемо варіювати, то елементи цієї множини не виходитимуть за визначені межі (фінітна точка зору). З огляду на це замість звичного запису «порівняння за модулем», використовуватимемо позначення $\underline{\underline{\mathcal{Z}_{m,n}}}$ — «рівно в кільці лишків», що скорочує запис операцій за модулем. Тут m, n — верхня і нижня межі цієї множини відповідно, до того ж якщо нижня межа дорівнює 1, то цей індекс не задіюватимемо. Природно, що в цьому разі класи лишків немає сенсу розглядати.

Зауважимо, що для множини (1) в цій інтерпретації

$$\sup \mathbf{Z}_m = m, \ \max\{z_i\} = m-1, \ \operatorname{int} \mathbf{Z}_m = 1, \min\{z_i\} = 0 \ \text{ і } \ \forall z_i = (\operatorname{int}(z)) \bmod m. \tag{14}$$

ТЕОРЕТИКО-ЧИСЛОВЕ ПРЕДСТАВЛЕННЯ СИСТЕМ ЧИСЛЕННЯ

У роботах [3, 4] доведено теорему, що має фундаментальне значення для теоретико-числових перетворень. Сформулюємо її без доведення.

Нехай алгебра виду $\mathcal{Z}_m =< \mathbf{S}, +, \cdot, \mathbf{0}, \mathbf{1} >$, де $0 \neq 1$, а $\mathbf{S} \in \mathbf{Z}$ — множина, що має $\sup \mathbf{S} = m$, $\inf \mathbf{S} = 1$, $\min\{z_i\} = 0$ (стосовно (11), (12), що являє собою кільце лишків з одиницею, в якому аргументами задана показникова функція $y = s^x$). Тоді для $\forall s \in \mathbf{S}$, $\forall p \in \mathbf{N}$, $\forall x = \overline{0, N}$ справедлива така залежність у кільці лишків $\mathcal{Z}_m$:

$$s^{(x) \bmod N} \underline{\underline{\mathcal{Z}_m}} (s^x) \bmod m, \tag{15}$$

де $m = \sum_{t=0}^{N-1} s^t$ — модуль кільця лишків $\mathcal{Z}_m$.

Ця теорема дає змогу формулювати узагальнене теоретико-числове перетворення з доведанням його основних теорем [5] у вигляді

$$X(k) \underline{\underline{\mathcal{Z}_m}} \sum_{i=0}^{N-1} x(i) s^{-(ki) \bmod N}, \tag{16}$$

$$x(i) \underline{\underline{\mathcal{Z}_m}} \frac{1}{N} \sum_{k=0}^{N-1} X(k) s^{(ki) \bmod N}, \tag{17}$$

де $x(i)$ — оригінал, $X(k)$ — зображення, $m = \sum_{t=0}^{N-1} s^t$ — модуль кільця лишків $\mathcal{Z}_m$.

Для теоретико-числового представлення систем числення послідовність оригінала $x(i)$ повинна бути зваженою послідовністю (вікном) вигляду [5]

$$w(i) = \{0, 1, 0, \ldots, 0\} \tag{18}$$

для того, щоб з цієї послідовності виокремити одне число. З огляду на вимоги систем числення до алфавіту цифр у розрядах (цілі числа, які не перевищують основи) уведемо в перетворення ще дві операції, що обмежують зверху і знизу члени послідовності оригіналу, операції виду (*) mod s, int (*) і представляють грані множини алфавіту цифр, унаслідок чого матимемо

$$X(k) \overset{Z_m}{=} \left\{ \text{int} \left[\sum_{i=0}^{N-1} x(i) \cdot w(i) s^{-(ki) \bmod N} \right] \right\} \bmod s. \tag{19}$$

Проаналізуємо вираз (19). З виразу (18) випливає, що $w(i) = 1$ для $i = 1$ і $w(i) = 0$, якщо $i \neq 1$. Тоді $x(i)w(i) = x(1)$, якщо $i = 1$, і $x(i)w(i) = 0$, якщо $i \neq 1$; вираз (19) перепишемо у вигляді

$$X(k) \overset{Z_m}{=} \left\{ \text{int} \left[\sum_{i=1}^{N-1} x(1) \cdot w(1) s^{-(k) \bmod N} \right] \right\} \bmod s. \tag{20}$$

Далі вважатимемо, що в (20) $s = p$, де p — основа системи числення, $X(k) = a_k$, $x(1) = A$, де A — число, що належить до обмеженої множини, для якої $\sup \mathbf{S} = \sum_{t=0}^{N-1} p^t$ (тобто верхня межа цієї множини збігається з модулем кільця лишків). Тоді отримаємо

$$a_k \overset{Zm}{=} \{\text{int}[Ap^{-(k) \bmod N}]\} \bmod p = \{\text{int}_{p^{(k) \bmod N}}[A]\} \bmod p. \tag{21}$$

Отже, одержано формулу обчислення цифр розрядів тієї чи іншої системи числення. Зауважимо, що $\lim\limits_{p \to \infty} (\text{int}_p(A))$ являє собою межу цілісності A.

На підставі (17) зворотне перетворення запишемо у вигляді

$$A = \sum_{k=0}^{N-1} a_k p^{(ki) \bmod N}, \quad i = 1. \tag{22}$$

Цю пару перетворень назвемо Р-перетвореннями, звертаючи увагу на їхній зв'язок з системами числення. Процес отримання цифр у розрядах системи числення з основою p можна ілюструвати матричним способом у такому вигляді:

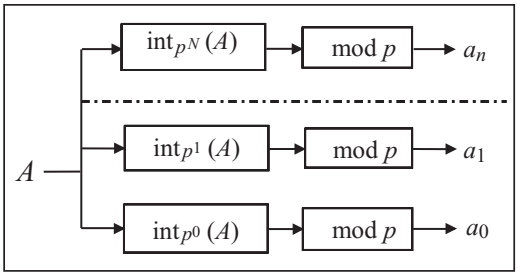

Рис. 1. Функціональна схема обчислень за виразом (21)

$$\text{int} \left[\begin{vmatrix} 1 & 1 & 1 \\ 1 & 1/p^0 & 1/p^1 \\ 1 & 1/p^1 & 1/p^0 \end{vmatrix} \times \begin{vmatrix} 0 \\ A \\ 0 \end{vmatrix} \right] \bmod p = \begin{vmatrix} a_0 \\ a_1 \\ a_2 \end{vmatrix} \tag{23}$$

або функціональною схемою (див. рис. 1).

З рис. 1 видно, що обчислення цифр виконується незалежно одне від одного у розрядах тієї чи іншої системи числення.

Стосовно алгебри система числення являє собою оболонку арифметичного простору, побудованого над кільцем лишків за модулем, який дорівнює основі системи числення. (Докладніше теоретико-числове зображення систем числення викладено у роботі [3].)

ФАКТОРИЗАЦІЯ МЕТОДОМ ПОСЛІДОВНИХ НАБЛИЖЕНЬ

Наведемо спочатку деякі факти стосовно систем числення, які причетні до факторизації чисел. Це ознаки ділення на будь-яке довільне число. Однією з таких ознак є ділення на основу системи числення, в якій саме число і представлено. Це свідчить про те, що якщо число C перебуває в системі числення з основою p, то це число ділиться на число p в тому разі, коли цифра наймолодшого розряду цього числа дорівнює нулю. Тоді

$$C = \sum_{i=1}^{n} c_i \cdot p^i = p \sum_{i=0}^{n-1} c_i \cdot p^i. \qquad (24)$$

Запишемо число C в дворозрядній системі числення:

$$C(p) = x * y = \text{int}_p (C) \cdot p^1 + [(C) \bmod p] \cdot p^0) = a_1 \cdot p^1 + a_0 \cdot p^0 = a_1 \cdot p + a_0, \quad (25)$$

де a_1 — цифра старшого розряду, a_0 — цифра молодшого розряду, p — основа системи числення, в якій представлено число, що факторизується, природно, що $p_1 = p$, а $p_0 = 1$.

Сутність факторизації у цьому разі полягатиме в пошуку такої системи числення, в якій основа дорівнюватиме одному з чисел, що складає число, яке факторизується, наприклад числу x з виразу (25). Тоді це число буде представлено добутком основи системи числення p на цифру старшого розряду a_1, тобто $C = a_1 \cdot p$, а молодший член виразу дорівнюватиме нулю. Отже, $a_0 = 0$ є критерієм пошуку потрібної системи числення.

Сам процес факторизації можна представити ітераційною процедурою за виразом (24) з початковими умовами $p = \text{int}\sqrt{C}$. Спочатку обчислимо вираз (25) і отримаємо значення цифр в розрядах цієї системи числення. Далі запозичимо одиницю із старшого розряду в молодший розряд за правилами арифметики в системах числення з урахуванням $p^0 = 1$ і $p^1 = p$, отримаємо

$$C(p) = a_1 \cdot p + a_0 = (a_1 - 1) \cdot p + p + a_0, \qquad (26)$$

де $C(p)$ — факторизовне подане число в системі числення з основою p.

Вочевидь, що $a_1 - 1 < p$, однак $p + a_0 > a_1 - 1$, тоді $p + a_0$ можна зобразити як $a_1 - 1 + r$, де r — деякий залишок від віднімання: $r = p + a_0 - a_1 - 1$. Підставимо цей вираз в (26) і приведемо подібні члени:

$$C(p+1) = (a_1 - 1) \cdot p + a_1 - 1 + r = (a_1 - 1) \cdot (p + 1) + r. \qquad (27)$$

Якщо $r > a_1 - 1$, то обчислення за виразом (27) слід повторювати, доки r не стане меншим за $a_1 - 1$. Тоді отримаємо число, що факторизується, в новій системі числення з цифрою в старшому розряді, зменшеною на одиницю, з новою основою, що дорівнює $p + \text{int}_{(a_1 - 1)} (p + a_0)$, і молодшим розрядом, що дорівнює $(p + a_0) \bmod (a_1 - 1)$:

$$C(p + \text{int}_{(a_1-1)}(p + a_0)) =$$

$$= (a_1 - 1) \cdot (p + \text{int}_{(a_1-1)}(p + a_0) + (p + a_0) \bmod (a_1 - 1). \quad (28)$$

Ітерацію, що задається виразами (27) і (28), слід виконувати, доки не буде отримано значення молодшого розряду одержуваної нової системи числення, яке дорівнюватиме нулю. Зауважимо, що вирази (27) і (28) записано для кроку, який дорівнює одиниці. Вочевидь, що крок може не відповідати одиниці. Тоді вирази для ітерації остаточно запишемо у вигляді

$$C(p) = a_1 \cdot p + a_0 = (a_1 - k) \cdot p + k \cdot p + a_0, \quad (29)$$

$$C(p + \text{int}_{(a_1-k)}(k \cdot p + a_0)) =$$

$$= (a_1 - k) \cdot (p + \text{int}_{(a_1-k)}(k \cdot p + a_0) + (k \cdot p + a_0) \bmod (a_1 - k).$$

Граф обчислень за цим методом представлено на рис. 2.

Приклад 1. Нехай $C = 57$ — число, яке потрібно факторизувати. Для першого наближення виберемо $p = 8$. Це найбільш придатне число, яке можна визначити, оскільки як середнє геометричне воно вочевидь належить інтервалу (a, b), бо $C = a * b$. Результати обчислень за різними початковими умовами зведено в табл. 1 (остаточний результат факторизації позначено жирним шрифтом).

Проаналізуємо ефективність методу. З табл. 1 видно, що його ефективність залежить від початкових умов. Якщо вибрані початкові умови дорівнюють одиниці, то виконується умова $p > a_1 > a_0$, а крок дорівнює 1; отже, кількість ітерацій складає

$$K_i = \text{int}\sqrt{C} + 1 - b, \quad (30)$$

де b — менший із співмножників факторизовного числа C.

Якщо обрано початкові умови, що дорівнюють двом, то кількість ітерацій скорочується вдвічі, тобто

$$K_i = (\text{int}\sqrt{C} + 1 - b) / 2, \quad (31)$$

оскільки завжди можна виконати першу ітерацію з кроком 1, а інші ітерації — з кроком 2. Має сенс факторизовувати тільки непарні складені числа унаслідок того, що парні числа тривіально після поділу на два перетворюються в непарні числа. Отже, для початкових умов 3 крок k вибрано вірно, тому що для факторизації знадобилася лише одна ітерація. Зауважимо, що на інтервалі (x, y) є число, величина якого дорівнює середньоарифметичному значень чисел x і y. Для початкових умов 4 кількість ітерацій складає

$$K_i = (\text{int}_{((a+b)/2)}C - b) / 2, \quad (32)$$

що зазвичай менше, ніж для початкових умов 2, оскільки

$$\text{int}_{((a+b)/2)}C < \text{int}\sqrt{C} + 1.$$

З аналізу випливає, що ефективність методу суттєво залежить від інтервалу (x, y).

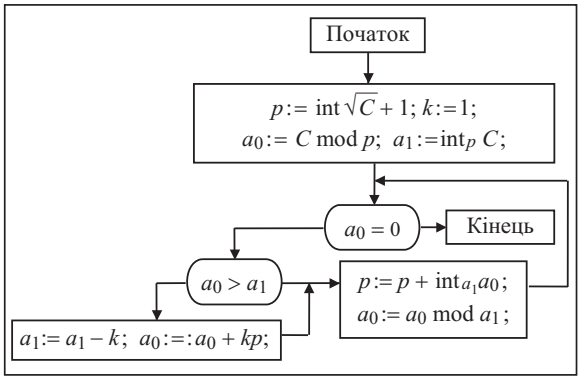

Рис. 2. Граф алгоритму факторизації складеного числа

Таблиця 1

Номер ітерації	Результати обчислень за початковими умовами			
	k	$a_1 := a_1 - k$	$a_0 := (kp + a_0) \bmod a_1$	$p := p + \text{int}_{a_1} a_0$
Початкові умови 1: $k = 1$, $a_1 = 7$, $a_0 = 1$, $p = 8$				
1	1	$7 - 1 = 6$	$(8 * 1 + 1 = 9) \bmod 6 = 3$	$8 + \text{int}_6 9 = 9$
2	1	$6 - 1 = 5$	$(9 * 1 + 3 = 12) \bmod 5 = 2$	$9 + \text{int}_5 12 = 11$
3	1	$5 - 1 = 4$	$(11 * 1 + 2 = 13) \bmod 4 = 1$	$11 + \text{int}_4 13 = 14$
4	1	$4 - 1 = \mathbf{3}$	$(14 + 1 = 15) \bmod 3 = \mathbf{0}$	$14 + \text{int}_3 15 = \mathbf{19}$
Початкові умови 2: $k = 2$, $a_1 = 7$, $a_0 = 1$, $p = 8$				
1	2	$7 - 2 = 5$	$(2 * 8 + 1 = 17) \bmod 5 = 2$	$8 + \text{int}_5 17 = 11$
2	2	$5 - 2 = \mathbf{3}$	$(2 * 11 + 2 = 24) \bmod 3 = \mathbf{0}$	$11 + \text{int}_3 24 = \mathbf{19}$
Початкові умови 3: $k = 4$, $a_1 = 7$, $a_0 = 1$, $p = 8$				
1	–	$7 - 4 = \mathbf{3}$	$(4 * 8 + 1 = 33) \bmod 3 = \mathbf{0}$	$8 + \text{int}_3 33 = \mathbf{19}$
Початкові умови 4: $k = 2$, $a_1 = 5$, $a_0 = 2$, $p = 11$				
1	2	$5 - 2 = \mathbf{3}$	$(2 * 11 + 2 = 24) \bmod 3 = \mathbf{0}$	$11 + \text{int}_3 24 = \mathbf{19}$

Тобто від різниці між числами, які складають факторизовне число, залежить різниця між основою системи числення p і меншим складеним числом цього числа C. Зменшити цей інтервал можливо шляхом множення числа C на відоме число t з інтервалу $(\text{int}\sqrt{C} + 1, 2)$. Тоді $C = a \cdot b \cdot t = a \cdot (b \cdot t)$ і кількість ітерацій у цьому разі складає

$$K_u = (\text{int}\sqrt{C \cdot t} + 1 - b \cdot t) / 2 = (\text{int}\sqrt{C} \cdot \sqrt{t} + 1 - b \cdot t) / 2 . \qquad (33)$$

З виразу (33) випливає, що основа системи числення збільшується на величину $\sqrt{t}$, тоді як b збільшується в t раз:

$$\text{int}\sqrt{C} + 1 - b > \text{int}\sqrt{C} \cdot \sqrt{t} + 1 - b \cdot t . \qquad (34)$$

Отже, немає сенсу отримувати корінь з числа, що факторизується, оскільки метод працює з будь-якими початковими умовами. Тому початкові умови можна визначити з того факту, що число в двійковій системі числення представлено відомою розрядною сіткою 2^n. Звідси випливає, що в початкових умовах основа системи числення дорівнюватиме $\sqrt{2^n} \approx 2^{n/2}$. Слід зазначити, що в цьому методі ми оперуємо даними значно меншої довжини, ніж саме факторизовне число. І, нарешті, запропонований метод факторизації чисел належить до класу відомих у математиці ітераційних методів послідовних наближень. Наприклад, в арифметиці він використовується під час ділення, множення або добування квадратного кореня.

Зауважимо також, що запропонований метод дає змогу факторизувати не тільки числа, складені з двох співмножників, але і необмеженої їхньої кількості:

$$C = \prod_{i-1}^{n} a_i . \qquad (35)$$

Для цього ітераційну процедуру розглядуваного методу потрібно виконати в межах зміни основи числення від $\text{int}\sqrt{C} + 1$ до $C / 2$ з кроком 1, якщо число парне, а далі — з кроком 2. За один прохід у кожному разі, коли $a_0 = 0$ з (29), будуть визначені можливі пари добутків для (34) разом з шуканими:

$$a_1 * (a_2 a_3 a_4 a_5 \ldots.),\ a_2 * (a_1 a_3 a_4 a_5 \ldots.),\ a_3 * (a_2 a_1 a_4 a_5 \ldots.),$$
$$a_4 * (a_1 a_3 a_1 a_5 \ldots.),\ldots$$

Зауважимо, що у разі, коли отриманий в результаті ітерацій вираз (35) дорівнюватиме $C(p) = 1 * C + 0$, де $p = C$, $a_0 = 0$, і водночас a_0 набуватиме значення, яке дорівнюватиме нулю тільки один раз за всю процедуру, це є достовірним тестом на простоту цього числа.

Приклад 2. Нехай потрібно знайти всі співмножники числа $C = 3 * 19 * 5 = 285$. Тоді $\text{int}\sqrt{C} + 1 = 16$. Результати факторизації наведено в табл. 2.

Таблиця 2

Номер ітерації	Результати факторизації за початковими умовами k=1, a_1=7, a_0 =1, p=8				
	k	$a_1 := a_1 - k$	$a_0 := (kp + a_0) \bmod a_1$	$p := p + \text{int}_{a_1} a_0$	Результат
1	1	$16 - 1 = \mathbf{15}$	$(17 + 13 = 30) \bmod 15 = \mathbf{0}$	$17 + \text{int}_{15} 30 = \mathbf{19}$	$\boldsymbol{a_0}$ **= 0, 15*19**
2	2	$15 - 2 = 13$	$(2 * 19 = 38) \bmod 13 = 12$	$19 + \text{int}_{13} 38 = 21$	
3	2	$13 - 2 = 11$	$(2 * 21 + 12 = 54) \bmod 11 = 10$	$21 + \text{int}_{11} 54 = 25$	
4	2	$11 - 2 = 9$	$(2 * 25 + 10 = 60) \bmod 9 = 6$	$25 + \text{int}_9 60 = 32$	
5	2	$9 - 2 = 7$	$(2 * 31 + 6 = 68) \bmod 7 = 5$	$32 + \text{int}_7 68 = 41$	
6	2	$7 - 2 = 5$	$(2 * 40 + 5 = 85) \bmod 5 = \mathbf{0}$	$40 + \text{int}_5 85 = 57$	$\boldsymbol{a_0}$ **= 0,** 5*57
7	2	$5 - 2 = 3$	$(2 * 57 = 114) \bmod 3 = \mathbf{0}$	$57 + \text{int}_3 114 = 95$	$\boldsymbol{a_0}$ **= 0, 3*95**
8	2	$3 - 2 = 1$	$(2 * 95 = 190) \bmod 1 = \mathbf{0}$	$95 + \text{int}_1 190 = 285$	$\boldsymbol{a_0}$ **= 0, 1*285**

Таким чином, факторизація чисел становить лише арифметичну операцію, в якій не брали до відома властивості чисел, їхню простоту, закони розподілу простих чисел тощо, а застосовували тільки відомості про системи числення, що є типовим для арифметичних операцій. Отже, факторизацію чисел відносять однозначно до P-класу складності алгоритмів. Проста реалізація цього методу засобами обчислювальної техніки дає змогу увести до складу команд сучасних комп'ютерів операцію факторингу чисел нарівні з добуванням квадратного кореня, діленням чисел тощо, а це дозволить ефективно оперувати добутками виду (35).

Зауважимо, що подвійність добутку двох чисел у кільці лишків (зірковий добуток [6, 7]) випливає з таких простих перетворень:

$$a = m - a`,\ b = m - b`,\ C = a * b = (m - a`) * (m - b`) = m^2 - m(a` + b`) + a` * b`,$$
$$\begin{cases} a` \times b` = C \bmod m, \\ (a` + b`)m = \text{int}\,(C / m), \end{cases} \tag{36}$$

де $a` = m - a$, $b` = m - b$, за умови, що $a` * b` < m$, і являє собою подвійність запису числа в дворозрядній системі числення з основою m, в якій цифра молодшого розряду a_0 дорівнює $a` \times b` = C \bmod m$, а цифра старшого розряду дорівнює $a_1 = a` + b` = \text{int}\,(C / m)$. Вочевидь, що розв'язання системи рівнянь (36) призводить до квадратного рівняння. Однак розв'язувати систему не має сенсу, оскільки перше рівняння є добутком двох малих чисел і його можна факторизувати за допомогою запропонованого методу послідовних наближень. А це свідчить про те, що існують способи прискорення факторизації чисел зазначеним методом.

ПОКАЗНИКОВА ФУНКЦІЯ І ДИСКРЕТНИЙ ЛОГАРИФМ

Розглянемо рівняння

$$a^x \equiv b \bmod m$$

у прийнятих нами позначеннях

$$a^x \overset{Z_m}{=} b \text{ або у вигляді } \log_a b \overset{Z_m}{=} x\ , \tag{37}$$

де x — дискретний логарифм.

З іншого боку, малу теорему Ферма $a^x \equiv 1 \bmod m$, де $x-1=m$, у прийнятих позначеннях можна записати у вигляді [8, 9]

$$a^x \overset{Z_m}{=} 1 \text{ або } \log_a 1 \overset{Z_m}{=} x\ . \tag{38}$$

Вирази (37) і (38) стосовно обчислень не відрізняються між собою, хіба що константою, і вочевидь, що існує один і той же алгоритм їхніх обчислень. Тоді запишемо (37) і (38) у такий спосіб:

$$a^x - b \overset{Z_m}{=} 0\ ,\ a^x - 1 \overset{Z_m}{=} 0\ .$$

Для розв'язування цих рівнянь потрібно поділити їхню праву частину за модулем m. Однак традиційне ділення зазвичай починають з боку старших розрядів, що призводить до перебору значень x, оскільки x і є невідомою змінною, яку ми шукаємо. Тоді, змінивши алгоритм ділення, будемо ділити, починаючи з боку молодших розрядів. Вочевидь, це можливо, якщо числа подати в системі числення за основою a таким чином:

$$a^x - 1 = (a-1)a^x + (a-1)a^{x-1} + \ldots + (a-1)a^1 + (a-1)a^0\ . \tag{39}$$

Із запису (39) випливає, що всі цифри в розрядах цієї системи числення однакові, що є важливим аргументом. Далі сформулюємо, власне, сам алгоритм ділення. У результаті ділення виразу (39) на модуль матимемо

$$(a^x - 1)/m \overset{Z_m}{=} 0\cdot a^x + 0\cdot a^{x-1} + \ldots + 0\cdot a^1 + 0\cdot a^0\ . \tag{40}$$

Зазвичай ділення здійснюємо відніманням дільника з діленого таким чином, щоб послідовно старші розряди приймали значення нуль. Однак внаслідок того, що виникає запозичення одиниці зі старшого розряду в молодший, яка спотворює результат у старшому розряді, то на практиці як умова використовується знак числа. Під час ділення, починаючи з молодшого розряду, запозичень не відбувається, тому такою умовою можна скористатися, перетворюючи кожен раз у нуль цифри молодшого розряду.

Таким чином, ділення представлятиме ітераційну процедуру виокремлення дільника з діленого доти, доки цифра молодшого розряду не дорівнюватиме нулю. Далі проводиться зсув дільника на один розряд і шляхом виокремлення дільника з діленого звертаємо в нуль цифру наступного за молодшим старшого розряду. Ці ітерації слід продовжувати, доки не поділимо все число, тобто доки не отримаємо вираз (40).

Приклад 3. Нехай потрібно обчислити дискретний трійковий логарифм числа 13 за модулем 17:

$$x \overset{Z_m}{=} \log_3 13.$$

Вибираємо систему числення з основою 3 і формуємо суму з виразу (39):

$$3^x - 1 = 2 \cdot 3^x + 2 \cdot 3^{x-1} + \ldots + 2 \cdot 3^1 + 2 \cdot 3^0. \qquad (41)$$

У трійковій системі це число має запис 22 ... 22. Уявімо також у трійковій системі числа 13 = 111 і 17 = 122. Далі віднімемо з виразу (41) число 13: $222222 - 111 = 222111$ і поділимо його на 17 стовпчиком:

2	2	2	1	1	1
		–	1	2	2
2	2	1	2	1	2
		–	1	2	2
2	2	1	0	2	**0**
	–	1	2	2	
2	2	2	1	**0**	
–	1	2	2		
2	1	2	2		
–	1	2	2		
2	**0**	**0**	**0**		

Отже, з одержаного в межах подання числа 17 (3-й розряд) отримано нулі, тому операцію слід завершити. У результаті отримано число 00000, яке складається з п'яти нулів, а відповідно до малої теореми Ферма $a^{p-1} = 1 \bmod p$ маємо $x = 4$, а також

$$\log_3 13 \overset{\mathbb{Z}_m}{=\!=} 4.$$

Вочевидь, що таким чином операцією ділення можна знайти і період показникової функції. Звідси випливає, що для обчислення періоду показникової функції і дискретного логарифма потрібна лише одна операція ділення чисел. Це дає змогу стверджувати, що складність алгоритмів обчислення періоду показникової функції та дискретного логарифма, а також факторизації чисел методом послідовних наближень (теж складає одну операцію) можна віднести до P-класу складності. Зауважимо, що в систему команд сучасних комп'ютерів слід ввести (крім команди факторизації чисел) обчислення періоду показникової функції та дискретного логарифмування, враховуючи простоту їхнього обчислення. Процедуру ділення, починаючи з молодшого розряду, можна економізувати, використовуючи двійково-трійкову систему числення та парність числа.

ВИСНОВКИ

Запропоновано представлення систем числення теоретико-числовими перетвореннями. Це подання дозволило створити кращий наразі алгоритм факторизації складових чисел. Показано, що факторизація чисел є звичайною арифметичною операцією, яка, як і ділення чисел, ґрунтується на методі послідовних наближень. Проста арифметична дія дає змогу легко реалізувати цей алгоритм окремою командою в будь-якій системі команд сучасного комп'ютера. Алгоритм також дає змогу факторизувати будь-який скінченний добуток чисел, що відкриває нові можливості використання цих добутків, а також добувати цілочисельне значення квадратного кореня, оскільки квадрат числа є тим же добутком, тільки з рівними співмножниками. Показано

також, що обчислення періоду показникової функції та дискретного логарифма зводиться до однієї звичайної операції ділення. Усі зазначені обчислення не створюють ніяких труднощів, оскільки належать до P-класу складності.

СПИСОК ЛІТЕРАТУРИ

1. Ишмухамедов Ш.Т., Рубцова Р.Г. О сложности задачи факторизации натуральных чисел. Казань: Вестник ТГГПУ. 2007. № 2–3. С. 4–6.
2. Семотюк М.В. Об аналитическом методе факторизации составных чисел. Киев: Институт кибернетики им. В.М. Глушкова НАН Украины. *Комп'ютерні засоби, мережі та системи.* 2013. № 12. С. 5–10.
3. Семотюк М.В. Заметки по машинной алгебре. Киев: Сталь, 2012. 250 с.
4. Семотюк М.В. Обобщенное теоретико-числовое преобразование. Киев: Институт кибернетики им. В.М. Глушкова НАН Украины. 1994. 29 с. (Препр. / НАН Украины, Институт кибернетики им. В.М. Глушкова, 94–8).
5. Семотюк М.В. Теоретико-числовые представления систем счисления. Киев: *УСиМ*, 2004. № 5. С. 36–42.
6. Ван дер Варден Б.Л. Алгебра. Москва: Наука, 1979. 624 с.
7. Ногин В.Д. Введение в математический анализ. СПб.: Санкт-Петерб. гос. политехн. ун-т., 1994. 62 с.
8. Olds C.D., Lax A., Davidoff G.P. The geometry of numbers. Washington, DC: Mathematical association of America, 2000. ISBN 0-88385-643-3.
9. Коблиц Н. Курс теории чисел и криптографии. Москва: Научное издательство ТВП, 2001. 254 с.

M.V. Semotiuk

NUMBER-THEORETICAL METHODS FOR FACTORIZATION OF COMPOZITE NUMBERS AND CALCULATION OF THE DISCRETE LOGARITHM

Abstract. The article is devoted to a new application of number-theoretic transformations. Representation of number systems by these transformations allows creating fundamentally new and efficient algorithms for factorizing numbers, calculating the period of the exponential function and the discrete logarithm. The factorization algorithm allows you to decompose any finite product into factors in one pass, and is also an exact test of the simplicity of numbers. Based on the representation of number systems by number-theoretic transformation, these algorithms have no analogs in the world, since they only use simple arithmetic operations. Information about the simplicity of numbers or other properties of numbers is not applied; therefore, factorization of numbers, calculations of the period of the exponential function and of the discrete logarithm are simply arithmetic operations, are performed in finite time, and belong to the P-class of complexity.

Keywords: set, faces of a set, algebra, residue ring, modulus, axiomatics of integers, number-theoretic transformation, number system, radix, factorization, arithmetic operation, exponential function period, discrete logarithm.

Надійшла до редакції 07.07.2021