УДК 519.6:519.712.3:510.52

Я.В. Зінченко

МЕТОДИ ОБЧИСЛЕННЯ ДОБУТКУ БАГАТОРОЗРЯДНИХ ЧИСЕЛ ТА ЇХ ОПТИМІЗАЦІЯ. ЧАСТИНА ІІ

Проведено аналіз відомих методів обчислення добутку багаторозрядних чисел, укладено їх класифікацію, наведено апріорні оцінки обчислювальної складності. Визначено області ефективного використання методів, дані рекомендації щодо їх застосування при розв'язанні прикладних задач. Здійснено пошук можливості та шляхів оптимізації методів. Розглянуто питання розробки адаптивного методу обчислення добутку багаторозрядних чисел.

Ключові слова: багаторозрядні числа, оптимізація, адаптивний метод.

На даний час існує багато прикладних задач, при розв'язанні яких активно використовується арифметика багаторозрядних чисел. До них належать задачі двоключової криптографії, спектрального і кореляційного аналізу та фільтрації цифрових сигналів, аеро- та гідродинаміки, розрахунку оболонок ядерних реакторів, моделювання фізичних, хімічних (біохімічних) процесів, обробки даних біофізичного експерименту та інші.

Однією з найбільш трудомістких операцій з багаторозрядними числами є операція обчислення добутку. На сьогоднішній день існує досить велика кількість методів його обчислення, кожен з яких має свою область ефективного застосування в залежності від області значень m (довжин чисел-співмножників), моделі обчислень, програмної чи апаратної реалізації. Усі ці методи є рекурсивними і засновані на зведенні множення багаторозрядних чисел до послідовності множень чисел з меншою кількістю розрядів.

Дана стаття є продовженням роботи [1]. Розглядаються два підходи до оптимізації за часом виконання на ЕОМ операції обчислення добутку багаторозрядних чисел: один ґрунтується на розробці більш ефективної структури програм, які реалізують алгоритми методу, інший – на залученні деяких резервів оптимізації методів. Викладення матеріалу буде вестись у відповідності з [1-8].

Оптимізація традиційного методу. Схема реалізації традиційного методу обчислення добутку двох багаторозрядних чисел a і b, яку було описано в [1], називається прямокутною, оскільки множення виконується за правилом множення многочленів, тобто спочатку обчислюються часткові добутки $a_i \cdot b_i$, а потім їх суми, що й дає необхідний результат t. Реалізація цієї схеми при множенні s-слівних чисел потребує $4s^2 + 3s$ звернень до пам'яті ЕОМ. Ця кількість може бути зменшена до $2s^2 + 2s$, якщо обчислювати суму кожного стовпчика на регістрах, а потім записувати у пам'ять:

$$\begin{array}{cccccc} & & a_2 & a_1 & a_0 \\ \times & & b_2 & b_1 & b_0 \\ \hline & & t_{02} & t_{01} & t_0 \\ & t_{12} & t_{11} & t_{10} & \\ + \; t_{22} & t_{21} & t_{20} & & \\ \hline t_5 & t_4 & t_3 & t_2 & t_1 \quad t_0 \end{array}.$$

Така схема реалізації називається діагональною [2].

Програма, що реалізує діагональну схему, називається MD (множення діагональне). Вона містить багато команд умовного переходу, оскільки повинна слідкувати за довжиною кожної діагоналі і їх кількістю. Якщо s фіксоване, то ці команди можна виключити, а також використати більш ефективні способи адресації команд. Такий прийом називається "розшивкою". Однак з ростом s довжина програми з "розшивкою" суттєво збільшується. В табл. 1 наведено довжини програм MDR (множення діагональне з "розшивкою") в кілобайтах для низки значень s. З таблиці видно, що використання "розшивки" доцільне при $s \le 13$. Для порівняння: програма MD займає 1 кБ пам'яті [3].

Таблиця 1

Залежність довжини програм MDR від розміру вхідних даних

s	4	5	7	8	12	13	30	52	64
$L_{MDR}(s)$	0,3	0,4	0,8	1,0	2,1	2,4	12,0	35,1	52,8

В табл. 2 для деяких значень s наведено оцінки коефіцієнтів прискорення часу роботи програми MDR відносно програми MD [3].

Таблиця 2

Залежність коефіцієнту прискорення програм MDR від розміру вхідних даних

s	4	5	7	8	12	13
K (MDR, MD)	0,54	0,54	0,56	0,56	0,57	0,57

Оптимізація методу Карацуби. Відомо, що побудова ефективніших за часом реалізації методів обчислення добутку багаторозрядних чисел базується на зведенні множення багаторозрядних чисел до послідовності множень чисел з меншою кількістю розрядів (однослівних чисел) і зменшенні їх кількості. Такий прийом називається "розщепленням". Його використання зменшує кількість однослівних множень, однак вимагає додаткових операцій додавання і віднімання. Затрати часу на виконання цих операцій при малих значеннях s можуть перекривати можливий виграш, тому існує деякий поріг для s, починаючи з якого застосування даного прийому стає вигідним. Так, одноразове використання "розщеплення" зменшує кількість однослівних множень приблизно на 25% і доцільне при $s \geq 12$. Дворазове "розщеплення" вигідне при $s \geq 20$. Істотна економія машинного часу при обчисленні добутку багаторозрядних чисел може бути досягнута при сумісному використанні "розщеплення" і "розшивки".

Якщо "розщеплення" застосовувати рекурсивно, то час обчислення добутку s-слівних чисел може бути зменшений з величини порядку s^2 до величини порядку $\approx s^{1,59}$.

Подальша оптимізація за часом реалізації методу обчислення добутку багаторозрядних чисел можлива за рахунок застосування прийому "повного розщеплення". При цьому задача обчислення добутку s-слівних чисел зводиться до множення однослівних чисел, додавання двохслівних і віднімання двохслівних (однослівних) чисел. Тоді множення чисел A і B може бути записане наступним чином:

$$A \cdot B = \sum_{i=0}^{s-1} a_i 2^{\omega i} \cdot \sum_{i=0}^{s-1} b_i 2^{\omega i} = \sum_{i=0}^{s-1}\sum_{j=0}^{s-1} a_i b_i 2^{\omega(i+j)} + \sum_{i=0}^{s-1}\sum_{j=0}^{i-1}\left(a_i - a_j\right)\cdot\left(b_j - b_i\right)2^{\omega(i+j)}. \qquad (1)$$

Складність операції обчислення добутку s-слівних чисел з використанням наведеної модифікації методу Карацуби дорівнює:

$$T(2s) = T(s) + C(s), \qquad (2)$$

де $T(s)$ – кількість множень однослівних чисел, $T(s) = \frac{1}{2}s^2 + \frac{3}{2}s - 1$; $C(s)$ – кількість додавань (віднімань) однослівних і двохслівних чисел, $C(s) = \frac{3}{2}s^2 + \frac{1}{2}s - 3$.

Із співвідношень видно, що дана модифікація методу Карацуби асимптотично в два рази ефективніша за традиційний метод за кількістю операцій множення однослівних чисел.

В табл. 3 для деяких значень s наведено коефіцієнти прискорення роботи програм, які реалізують метод Карацуби з "розщепленням" (k_u^1) і "повним розщепленням" (k_u^2), відносно програми MD. Порівняння цих коефіцієнтів підтверджує ефективність модифікації методу Карацуби з "повним розщепленням" [4].

Таблиця 3

Залежність коефіцієнтів прискорення роботи програм від розміру вхідних даних

s	64	128	256	512	1024
k_u^1	0,45	0,78	0,74	0,77	0,74
k_u^2	0,8016	0,7082	0,6737	0,6398	0,6317

Оптимізація "згорткових" методів. Відомо, що добуток двох багаторозрядних чисел (без врахування переносів) являє собою дискретну циклічну згортку співмножників. Оскільки обчислення такої згортки дає основний внесок в оцінку складності "згорткових" методів, то одним із основних шляхів їх оптимізації є пошук (розробка) ефективних алгоритмів згортки або ж модифікація вже існуючих. У зв'язку з цим В.К. Задіракою був запропонований метод обчислення добутку багаторозрядних чисел, заснований на використанні теореми про дискретну згортку двох функцій та оригінальної модифікації алгоритму швидкого перетворення Фур'є (ШПФ) з попередньою заготовкою елементів матриці перетворення, який має особливості і переваги у порівнянні з іншими відомими алгоритмами. Його застосування дозволяє суттєво зменшити кількість операцій при обчисленні згортки, а складність самого методу множення оцінюється як:

$$T^{\times} = 6K\log K + 8K,$$

$$C^{\pm} = 6K\log K + 5K,$$

де K – довжина дискретного перетворення Фур'є (ДПФ), $T^{\times}$ – кількість однослівних множень, $C^{\pm}$ – кількість однослівних додавань (віднімань).

Подальше зменшення часу реалізації даного методу на ЕОМ можливе за рахунок використання наступних резервів оптимізації:

1. При обчисленні ДПФ дійсного сигналу використовується надлишок даних (використовуються властивості ДПФ дійсних сигналів та їх взаємозв'язок, а також взаємозв'язок ДПФ розрядності N та $2N$), що дозволяє приблизно в два рази зменшити кількість операцій множення комплексних чисел.

2. Матриця перетворення Фур'є, елементами якої є передобчислені синуси та косинуси, зменшується в два рази за рахунок використання взаємозв'язку між значеннями синусів та косинусів: $\cos_{2i} = \sin_{2i+1}$, $\cos_{2i+1} = -\sin_{2i}$, $i = \overline{0,\ N/4-1}$, де N – розрядність згортки.

3. Вхідні послідовності (числа-співмножники) розбиваються на блоки довжиною 24 біти (три байти), що дозволяє максимально використовувати 63-бітну мантису співпроцесора та скоротити кількість операцій за рахунок зменшення довжини згортки у три рази.

4. Оброблювані дані перегруповуються за рахунок попарного розміщення елементів дійсної та комплексної частин коефіцієнтів ДПФ, що дозволяє на 11% підвищити швидкодію методу за рахунок збільшення операцій послідовного доступу.

5. Використовується динамічний розподіл пам'яті, що дозволяє збільшити розмір оброблюваних даних до 6144 байт та зменшити час їх обробки на 20%.

З урахуванням наведених резервів оптимізації складність методу обчислення добутку багаторозрядних чисел з ШПФ може бути оцінена як:

$$T^{\times} = 3K\log K + 7K, \qquad (3)$$

$$C^{\pm} = 3K\log K + 12K, \qquad (4)$$

де K – довжина ДПФ (кількість трьохбайтних блоків, на які розбиваються співмножники), $T^{\times}$ – кількість операцій множення, $C^{\pm}$ – кількість операцій додавання (віднімання).

В табл. 4 та табл. 5 для деяких довжин чисел-співмножників та засобів обчислювальної техніки наведено коефіцієнти прискорення роботи програм, які реалізують метод з ШПФ до оптимізації (k_u^0) і після оптимізації (k_u^1), відносно програми MD. Аналіз даних, які містяться в таблицях, показує, що наведена оптимізація дозволяє зсунути порогове значення n_p на 2052 байти (16416 біт) вправо і підтверджує ефективність оптимізованого методу [5].

Таблиця 4

Залежність коефіцієнту прискорення роботи програми від розміру вхідних даних та ПЕОМ

n	512	1024	2048	4096	8192	16384	32768
k_u^0 для ПЕОМ 166 МГц	16	10	5,5	3	1,7	0,99	0,56
k_u^0 для ПЕОМ 400 МГц	8,2	4,4	2,3	1,4	0,75	0,42	0,23

Таблиця 5

Залежність коефіцієнту прискорення роботи програми від розміру вхідних даних та ПЕОМ

n	768	1536	3072	6144	12288	24576	49115	98230
k_u^1 для ПЕОМ 166 МГц	4,5	2,4	1,6	1,01	0,7	0,4	0,23	0,14
k_u^1 для ПЕОМ 166 МГц	4,4	2,5	1,4	0,8	0,4	0,23	0,13	0,07

Оскільки використання алгоритму ШПФ для обчислення дискретної циклічної згортки пов'язане з деякими обчислювальними труднощами (витрати машинного часу на обчислення тригонометричних функцій та боротьба з помилками заокруглення при обчисленні $w_k = \exp(2\pi i / K)$, $i = \sqrt{-1}$), то В.К. Задіракою був запропонований метод множення багаторозрядних чисел, заснований на використанні ефективного алгоритму Пітассі обчислення згортки, реалізація якого виключає перехід в поле комплексних чисел. В основу вказаного алгоритму згортки покладено трансформацію просторів вхідних даних $N = 2^n$ в простори відповідних коефіцієнтів Уолша вимірності $2\cdot 3^{n-1}$ і їх комбінацій у вигляді сум і різниць. Для ефективного обчислення коефіцієнтів Уолша в ньому використовується алгоритм швидкого перетворення Уолша (ШПУ). Застосування даного алгоритму згортки дозволяє суттєво зменшити кількість операцій при обчисленні добутку багаторозрядних чисел, а складність самого методу множення оцінюється як:

$$Q_{FFW}^{\times} = 2\cdot 3^{n-1},$$

$$Q_{FFW}^{+} = 13\cdot 3^{n-1} + 2^{n+1}(n - 2{,}25) - 1,$$

де $Q^{\times}_{FFW}$ – кількість операцій множення, Q^{+}_{FFW} – кількість операцій додавання.

Подальша оптимізація даного методу за часом реалізації на ЕОМ можлива за рахунок удосконалення алгоритму Пітассі:

1. При обчисленні циклічної згортки розрядністю $N = 2^n$ операція обчислення коефіцієнтів Уолша з використанням алгоритму ШПУ замінюється на операцію обчислення цих коефіцієнтів з використанням алгоритму швидкого перетворення Хаара (ШПХ).

2. При обчисленні циклічної згортки розрядністю $N = 2^n$ на основі алгоритму ШПУ вводиться корегуючий вектор.

3. При обчисленні згортки розрядністю $N = k2^n$, k – непарне, згортка розрядністю $2k$, k – непарне, представляється двома згортками меншої розрядності k замість трьох згорток.

Розглянемо зазначені резерви оптимізації по-порядку.

Відомо, що кількість операцій додавання, необхідних для обчислення коефіцієнтів Уолша та Хаара векторів довжиною $N = 2^n$ з використанням швидких алгоритмів, дорівнює $n2^n$ та $2\left(2^n - 1\right)$ відповідно. З наведених нижче співвідношень, які пов'язують перетворення Уолша та Хаара, випливає, що перетворення Уолша можна замінити перетворенням Хаара. Останнє забезпечує економію кількості додавань (при $N = 2^n = 256$ приблизно в чотири рази) і, відповідно, більш високу швидкість обчислень. Ці співвідношення дають сімейство ортогональних перетворень, яке включає перетворення Уолша та Хаара. До цих перетворень відноситься один загальний алгоритм швидкого обчислення.

Позначимо матриці Хаара $\left[H_N\right]$ і Уолша-Адамара $\left[W_N\right]$ порядку N, рядки яких являють собою N функцій Хаара і Уолша, нормованих на $1/\sqrt{N}$. Розіб'ємо матриці $\left[H_N\right]$ і $\left[W_N\right]$ на $(n+1)$ прямокутних підматриць $\left[MH_N^k\right]$ і $\left[MW_N^k\right]$ розміром $\left(N \times 2^{k-1}\right)$, $k = 1, ..., n$. Матриця $\left[MH_N^0\right]$ являє собою перший рядок H^0, матриця $\left[MW_N^0\right]$ являє собою перший рядок W^0, а матриці $\left[MH_N^k\right]$ і $\left[MW_N^k\right]$ формуються з функцій Хаара й Уолша рангу r, причому $2^{k-1} \le r < 2^k$.

Між підматрицями $\left[MH_N^k\right]$ й $\left[MW_N^k\right]$ існує матричне співвідношення, яке їх зв'язує:

$$\left[MW_N^k\right] = \left[W_{2^{k-1}}\right] \cdot \left[S_{2^{k-1}}\right] \cdot \left[MH_N^k\right], \quad k = 1, ..., n, \tag{5}$$

де $\left[W_{2^{k-1}}\right]$ – упорядкована матриця Уолша-Адамара порядку 2^{k-1}, а $\left[S_{2^{k-1}}\right]$ – матриця перестановок порядку 2^{k-1}. Оскільки $\left[W_{2^{k-1}}\right]$ і $\left[S_{2^{k-1}}\right]$ симетричні й ортогональні, то є можливість одержати зворотне співвідношення:

$$\left[MH_N^k\right] = \left[W_{2^{k-1}}\right] \cdot \left[S_{2^{k-1}}\right] \cdot \left[MW_N^k\right], \quad k = 1, ..., n. \tag{6}$$

Задамо вектор V довжиною $N = 2^n$. Помноживши праві частини виразів (5) і (6) на вектор V, одержимо:

$$\begin{pmatrix} V_{W_{2^{k-1}}} \\ . \\ V_{W_{2^{k-1}}} \end{pmatrix} = \left[S_{2^{k-1}}\right] \cdot \left[W_{2^{k-1}}\right] \cdot \begin{pmatrix} V_{H_{2^{k-1}}} \\ . \\ V_{H_{2^{k-1}}} \end{pmatrix}, \tag{7}$$

$$\begin{pmatrix} V_{H_{2^{k-1}}} \\ . \\ V_{H_{2^{k-1}}} \end{pmatrix} = \left[W_{2^{k-1}}\right] \cdot \left[S_{2^{k-1}}\right] \cdot \begin{pmatrix} V_{W_{2^{k-1}}} \\ . \\ V_{W_{2^{k-1}}} \end{pmatrix}. \qquad (8)$$

Набори коефіцієнтів перетворених векторів, що з'являються у виразах (7) і (8), називаються зонами. Із співвідношень видно, що зона перетвореного вектора V_H визначає відповідну зону перетвореного вектора V_w. Ця властивість показує, що якщо вектор V апроксимується деякою підмножиною зон перетворених векторів V_H і V_w чи, зокрема, якщо ці вектори усікаються наприкінці зони, то після зворотних перетворень виходить вхідний наближений вектор V.

У співвідношеннях (7) і (8) відповідні зони зв'язані ортогональними перетвореннями. З теореми Парсеваля випливає, що енергії відповідних зон перетворених векторів однакові.

Таким чином, застосування співвідношень (5) і (6) призводить до того, що перетворення Хаара діє так само, як і перетворення Уолша, а кількість операцій додавання, необхідних для обчислення коефіцієнтів Уолша послідовності довжини $N = 2^n$, зменшується на величину $n2^n - 2\left(2^n - 1\right) = n2^n - 2^{n+1} + 2$. З урахуванням того, що при обчисленні згортки оброблюються дві вхідні послідовності, і для кожної з них виходить зазначений виграш, загальне зменшення кількості додавань складе $2^{n+1}(n-2)+4$.

Загальна кількість операцій додавання, необхідних для обчислення циклічної згортки удосконаленим алгоритмом Пітассі, дорівнює $13 \cdot 3^{n-1} - 1{,}5 \cdot 2^n - 4$, а складність методу обчислення добутку багаторозрядних чисел після оптимізації оцінюється як:

$$Z^{\times}_{FFH} = Q^{\times}_{FFW} = 2 \cdot 3^{n-1}, \qquad (9)$$

$$Z^{+}_{FFH} = 13 \cdot 3^{k-1} + 1{,}5 \cdot 2^k - 19, \qquad (10)$$

де $Z^{\times}_{FFH}$ – кількість операцій множення (залишається незмінною і дорівнює кількості операцій множення для методу-прототипу), Z^{+}_{FFH} – кількість операцій додавання.

Більш детальний опис методики удосконалення алгоритму Пітассі за кількістю операцій додавання, необхідних для обчислення циклічної згортки, наведено у [6].

В табл. 6 для деяких довжин чисел-співмножників наведено кількість операцій додавання Q^{+}_{FFW} та Z^{+}_{FFH}, необхідних для обчислення добутку багаторозрядних чисел вказаним методом до оптимізації та після його оптимізації. Аналіз даних, які містяться в таблиці, підтверджує ефективність оптимізованого методу [6].

Таблиця 6

Порівняння кількості додавань, необхідних для множення багаторозрядних чисел

n	5	6	7	8	9	10	11	12
m (біт)	256	512	1024	2048	4096	8192	16384	32768
Q^{+}_{FFW}	1228	3638	10692	31374	92204	271750	803476	2382782
Z^{+}_{FFH}	1082	3236	9650	28796	86042	257396	770690	2309036

В загальному вигляді алгоритм Пітассі обчислення циклічної згортки $R_N = X_N \otimes Y_N$, $N = 2^n$, може бути представлений наступними співвідношеннями [7]:

$$ER_N = EX_N \otimes EY_N + OX_N \otimes OY_N,$$
$$OR_N = EX_N \otimes OY_N + OX_N \otimes U(EY_N), \qquad (11)$$

де E, O, U – оператори ($E-Even$ (парний), $O-Odd$ (непарний), $U-Up$ (угору)): $(EX_N) = x_{2k}$, $(OX_N) = x_{2k+1}$, $k = \overline{0,\ N/2-1}$; $V_N = UX_N$, $v_k = x_{(k+1)_N}$, $k = \overline{0,\ N-1}$.

Якщо в алгоритм (11) ввести додатковий корегуючий вектор, то довжина відповідних послідовностей та обчислювальна складність щодо операцій однослівного додавання (віднімання) та множення зменшиться на 17%. В роботі [7] запропонована оптимізація проілюстрована на прикладі обчислення згортки довжиною $N = 4$ з використанням ШПУ. При цьому необхідно обчислювати корегуючий елемент t і достатньо виконати лише 5 операцій множення. При збільшенні розрядності згортки збільшується і кількість корегуючих елементів, які можуть бути представлені у вигляді корегуючого вектора, і пропонується простий спосіб його знаходження.

Введені додаткові оператори W, A, S ($W-Walsh$ (Уолш), $A-Add$ (додавання), $S-Substruct$ (віднімання)). Оператор W являє собою 1-й крок у перетворенні Уолша.

$$X_4 = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad Y_4 = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_4 \end{bmatrix}, \quad W_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}; \quad T_4 = \begin{bmatrix} 0 \\ -t \\ 0 \\ +t \end{bmatrix}, \quad R_4 = \frac{1}{4} W_4 \cdot \hat{A}_4 + T_4;$$

$$t = (x_1 - x_3)\cdot(y_0 - y_2), \quad \hat{A}_4 = (W_4 \cdot X_4)\cdot(W_4 \cdot Y_4).$$

В табл. 7 для деяких значень n наведено кількість операцій множення та додавання, необхідних для обчислення циклічної згортки різними алгоритмами: стандартним, алгоритмом на основі ШПФ, алгоритмом Пітассі на основі ШПУ, удосконаленим алгоритмом Пітассі на основі ШПХ і викладеним вище удосконаленим алгоритмом Пітассі на основі ШПУ. Аналіз даних, які містяться в таблиці, показує, що при $n \le 11$ останній алгоритм вимагає найменшої кількості операцій множення, що підтверджує його ефективність, і тому його доцільно використовувати для обчислення згорток довжиною

$N \le 2^{11} \le 2048$. Слід також зауважити, що $Q^{\times}_{FFT}$ і Q^{+}_{FFT} – це кількість операцій з плаваючою комою, а $Q^{\times}_{FWT}$ і Q^{+}_{FWT} – кількість операцій над цілими числами. Оскільки на сучасних комп'ютерах операції з плаваючою комою виконуються на порядок повільніше, ніж операції над цілими числами, виграш від застосування удосконаленого алгоритму може бути ще більшим.

Таблиця 7

Порівняння обчислювальної складності алгоритмів обчислення циклічної згортки

n	$Q^{\times}_{ST}$	Q_{FFT}		Q_{FFW}		Z_{FFH}		Q_{FWT}	
		$Q^{\times}_{FFT}$	Q^{+}_{FFT}	$Q^{\times}_{FFW}$	Q^{+}_{FFW}	$Z^{\times}_{FFH}$	Z^{+}_{FFH}	$Q^{\times}_{FWT}$	Q^{+}_{FWT}
12	16777216	221184	331776	354294	2378687	354294	2304941	295245	5175151
11	4194304	101376	152064	118098	801429	118098	768643	98415	1611219
10	1048576	46080	69120	39366	270727	39366	256373	32805	499499
9	262144	20736	31104	13122	91693	13122	85531	10935	154103
8	65536	9216	13824	4374	31119	4374	28541	3645	47271
7	16384	4032	6048	1458	10565	1458	9523	1215	14395
6	4096	1890	2592	486	3575	486	3173	405	4339
5	1024	720	1080	162	1197	162	1051	135	1287

Оцінка кількості обчислювальних витрат, необхідних для знаходження добутку багаторозрядних чисел за допомогою удосконаленого алгоритму Пітассі на основі ШПУ, має вигляд [7]:

$$Q^{\times}_{FWT} = 5 \cdot 3^{n-2}, \tag{12}$$

$$Q^{\pm}_{FWT} = n2^{n} - 11 \cdot 2^{n-2} + (2n+5)3^{n-1}, \tag{13}$$

де $Q^{\times}_{FWT}$ – загальна кількість операцій множення, $Q^{\pm}_{FWT}$ – загальна кількість операцій додавання (віднімання).

Згідно [8] циклічна згортка $R_N = X_N \otimes Y_N$ має наступні властивості: $UX_N \otimes UY_N = X_N \otimes Y_N$, $X_N \otimes UY_N = U(X_N \otimes Y_N)$, $UX_N \otimes Y_N = X_N \otimes DY_N$, де U, D – оператори (U – *Up* (угору), D – *Down* (до низу)): $V_N = UX_N$, $v_k = x_{\langle k+1 \rangle_N}$, $k = \overline{0,\ N-1}$; $V_N = DX_N$, $v_k = x_{\langle k+N-1 \rangle_N}$, $k = \overline{0,\ N-1}$. У загальному вигляді пропонуються формули обчислення згортки розрядністю $N = k2^n$, $n > 1$, k – непарне, $p = [k/2]$:

$$A_{N/2} = \left(EX_N + U^p(OX_N)\right) \otimes \left(EY_N + U^p(OY_N)\right),\ ER_N = 1/2\left(A_{N/2} + S_{N/2}\right), \tag{14}$$

$$S_{N/2} = \left(EX_N - U^p(OX_N)\right) \otimes \left(EY_N - U^p(OY_N)\right),\ OR_N = 1/2\left(A_{N/2} - S_{N/2}\right), \tag{15}$$

де E, O, U – оператори (E – *Even* (парний), O – *Odd* (непарний), U – *Up* (угору)): $(EX_N) = x_{2k}$, $(OX_N) = x_{2k+1}$, $k = \overline{0,\ N/2-1}$; $V_N = U^p(X_N)$, $v_k = x_{\langle k+p \rangle_N}$, $k = \overline{0,\ N-1}$.

В табл. 8 наведено кількість операцій множення, необхідних для обчислення циклічної згортки розрядністю $N = k2^n$, $n > 1$, $k = 3, 5, 7, 9$ [8].

Таблиця 8

Складність алгоритму обчислення циклічної згортки розрядністю $N = k2^n$

k	Кількість операцій множення $Q^{\times}(k)$ для обчислення згортки розрядністю k	Розрядність згортки $N = k2^n$	Кількість операцій множення $Q^{\times}(N)$ при обчисленні згортки розрядністю $N = k2^n$
3	4	$N = 3\times2^n$	$Q^{\times}(N) \le 8\times3^{n-1}$
5	10	$N = 5\times2^n$	$Q^{\times}(N) \le 20\times3^{n-1}$
7	16	$N = 7\times2^n$	$Q^{\times}(N) \le 32\times3^{n-1}$
9	19	$N = 9\times2^n$	$Q^{\times}(N) \le 38\times3^{n-1}$

Розробка адаптивного методу. Автором даної статті розглядається питання розробки адаптивного методу обчислення добутку багаторозрядних чисел, який враховував би не тільки область значень m (довжину чисел-співмножників), але й структуру багаторозрядних чисел (вид функцій) конкретної прикладної задачі. Адаптивність передбачає підбір підходящого базису, наприклад, Фур'є, Уолша, Хаара, і його адаптацію до кожного конкретного сигналу з метою забезпечення заданої точності якомога меншою кількістю членів ряду і прискорення процесів обробки.

Література

1. Зінченко Я.В. Методи обчислення добутку багаторозрядних чисел та їх оптимізація. Частина І / Я.В. Зінченко // Сучасний захист інформації. – 2013. – № 4. – С. 39-47.

2. Comba P. G. Exponentiation cryptosystems on the IBM PC // IBM Systems Journal. – 1990. – Vol. 29, No. 4. – P. 526-538.

3. Задирака В.К. О тестировании быстродействия алгоритмов и программ вычисления основных операций асимметричной криптографии / А.И. Березовский, В.К. Задирака, Л.Б. Шевчук // Кибернетика и системный анализ. – 1999. – № 5. – С. 59-66.

4. Задирака В.К. Оптимизация алгоритмов быстрого умножения больших чисел. I / В.К. Задирака, С.С. Мельникова, А.Н. Терещенко // Управляющие системы и машины. – 2006. – № 3. – С. 13-21.

5. Задирака В.К. Оптимизация алгоритмов быстрого умножения больших чисел. II / В.К. Задирака, С.С. Мельникова, А.Н. Терещенко // Управляющие системы и машины. – 2006. – № 4. – С. 23-32.

6. Зинченко Я.В. Метод умножения многоразрядных чисел асимметричных криптоалгоритмов, основанный на применении быстрого преобразования Хаара в операциях дискретной свертки / Я.В. Зинченко, А.В. Корнейко // Моделювання та інформаційні технології : Зб. наук. праць Інституту проблем моделювання в енергетиці НАН України. – К. : ІПМЕ, 2008. – Вип. 46. – С. 67-73.

7. Терещенко А.Н. Реализация операции умножения с использованием преобразования Уолша / А.Н. Терещенко, С.С. Мельникова, Л.А. Гнатив, В.К. Задирака, Н.В. Кошкина // Международный научно-технический журнал Проблемы управления и информатики. – 2010. – № 2. – С. 102–126.

8. Задирака В.К. Использование арифметики многоразрядных чисел в современных компьютерных технологиях решения задач трансвычислительной сложности / В.К. Задирака, А.Н. Терещенко // Научно-теоретический журнал Искусственный интеллект. – 2010. – № 3. – С. 712–728.

Надійшла 23.07.2014 р. Рецензент: д.т.н., проф. Єрохін В.Ф.