

Т.Б. МАРТИНЮК

Вінницький національний технічний університет, Вінниця, Україна,
e-mail: *martyniuk.t.b@gmail.com*.

Б.І. КРУКІВСЬКИЙ

Вінницький національний технічний університет, Вінниця, Україна,
e-mail: *smiletex11@gmail.com*.

ОСОБЛИВОСТІ ПАРАЛЕЛЬНОГО АЛГОРИТМУ СОРТУВАННЯ З ФОРМУВАННЯМ РАНГІВ

Анотація. Проаналізовано новий підхід до організації паралельного сортування масиву чисел з формуванням їхніх рангів. У процесі сортування реалізовано такі операції, як операція декремента для оброблення елементів числового масиву та операція інкремента для формування їхніх рангів. Запропоновано опис алгоритму паралельного сортування з формуванням рангів у базисі системи алгоритмічних алгебр (САА) Глушкова.

Ключові слова: система алгоритмічних алгебр, паралельне сортування, числовий зріз, ранг, маска.

ВСТУП

Практичний інтерес до вдосконалення методів і засобів сортування масивів даних [1], пов'язаний з широкою сферою їхнього застосування, яка охоплює, зокрема, пошукові системи INTERNET, системи управління базами даних (СУБД), оброблення сигналів і зображень (медіанна і рангова фільтрація) [2]. Крім того, сучасна елементна база (ПЛІС та оптоелектронні матриці смарт-пікселів) дає змогу апаратно реалізувати, наприклад, мережеві методи сортування значних масивів даних з прийнятною швидкістю [3].

Разом з тим як для програмної, так і для апаратної реалізації розроблюваних алгоритмів бажано наочно бачити їхній опис у компактному і простому вигляді. У цьому добре себе зарекомендували формалізовані описи широкого кола алгоритмів і, зокрема, системи алгоритмічних алгебр (САА) Глушкова [4–7]. Базис САА Глушкова [8], в свою чергу, розширюється модифікаціями [9], а процес паралельного програмування на його основі вдосконалюється [10–12]. У цій статті розглянуто особливості подання паралельного алгоритму в термінах САА Глушкова.

Мета роботи — підтвердження функціональних можливостей компактного опису в термінах САА Глушкова різновидів паралельних методів сортування.

ОСОБЛИВОСТІ ПОЗРІЗОВОГО ОБРОБЛЕННЯ МАСИВУ ЧИСЕЛ

Особливістю цього алгоритму сортування лінійного масиву чисел є паралельне позрізове оброблення. Існують кілька варіантів позрізового оброблення, де формується послідовність зрізів (slices). Наприклад, оброблення за розрядними зрізами використовують для паралельно-послідовного оброблення масивів даних в асоціативних процесорах [13–16].

Існує ще один різновид позрізового оброблення, де за зріз береться вектор різниці, що формується шляхом одночасного віднімання з усіх елементів поточного векторного масиву чисел найменшого (ненульового) з них. У роботі [17] такий зріз прийнято позначати як різницевий зріз. Цей принцип різницево-зрізового оброблення обґрунтовано і реалізовано під час моделювання порогового нейрона [17], а також описано в матеріалах САА Глушкова для мультиоброблення векторних масивів даних [18] і класифікації об'єктів на основі дискримінантних функцій [19].

Водночас необхідно відзначити, що використання оброблення за розрядними зрізами для сортування дає змогу формувати ранги (індекси) елементів відсортованого масиву [15, 16]. У цьому випадку в результаті сортування формується лінійний масив рангів. Кожен ранг розташований на позиції, що відповідає позиції елемента у вхідному масиві. Крім того, величина рангу відповідає позиції елемента у відсортованому масиві [20].

Сформовані ранги можна також вважати адресами, за якими зручно зчитувати з асоціативної пам'яті необхідний елемент масиву [13]. Особливо актуальним є застосування ранжування елементів під час їхнього сортування для медіанної фільтрації зображень [2, 21], а також для визначення екстремальних (мінімального / максимального) елементів у масиві чисел [13].

АЛГОРИТМ ПОЗРІЗОВОГО СОРТУВАННЯ ЛІНІЙНОГО МАСИВУ ЧИСЕЛ

У роботі [22] описано і досліджено модифікацію позрізового оброблення, в якому застосовується операція декременту (зменшення на одиницю) одночасно до всіх елементів масиву чисел з послідовним їх обнуленням. Саме момент фіксації ознаки обнулення певного елемента масиву використовується для формування результату асоціативного оброблення, тобто для пошуку екстремальних чисел і для пошуку за ключем серед елементів вхідного масиву чисел [22].

У цій роботі розглянуто процес сортування, в якому застосовано операцію декременту відносно елементів числового масиву, а також операцію інкремента (збільшення на одиницю) відносно рангів відповідних елементів. Отже, відбувається послідовне обнулення елементів вхідного масиву чисел $M = \{a_i\}$ і «вирушування» їхніх відповідних рангів $R = \{r_i\}$, $i = 1, n$. Процес запропонованого позрізового сортування з формуванням рангів містить такі операції:

- а) встановлення у початковий (одичний) стан рангів і елементів маски для всіх елементів масиву;
- б) декремент над усіма елементами масиву;
- в) маскуванню рангу відповідного обнуленого елемента масиву;
- г) інкремент над усіма незамаскованими рангами елементів масиву.

Основні умови (ознаки) цього процесу:

- а) принаймні один i -й елемент вхідного масиву в поточний момент часу дорівнює нулю, тобто $\exists a_i = 0$; $i = 1, n$;
- б) всі n елементів вхідного масиву в поточний момент часу дорівнюють нулю, тобто $\forall a_i = 0$.

Перша умова використовується для обнулення відповідного i -го елемента маски і маскуванню i -го рангу. Друга умова свідчить про завершення процесу сортування.

У табл. 1 наведено приклад сортування з ранжуванням масиву чотирьох чисел $\{2, 5, 3, 4\}$. У нульовому циклі в масиві рангів фіксується одиниця. Протягом першого циклу елементи масиву чисел зменшуються на одиницю (операція декременту), а елементи масиву рангів, в свою чергу, залишаються без змін. У другому циклі один з елементів масиву чисел обнулюється, а саме перший елемент. В результаті його ранг залишається незмінним (зі значенням 1), а ранги всіх інших чисел збільшуються на одиницю і приймають значення 2 (операція інкремента). Процес сортування відбувається доки в масиві чисел всі елементи не перетворяться на нулі, в результаті буде сформований масив рангів.

У табл. 1 виокремлено моменти, коли одне число з масиву обнулюється і маскується, залишаючись з тим же рангом; у той же час всі інші ранги збільшуються на одиницю. Процес сортування з ранжуванням виконано за п'ять

Таблиця 1. Процес сортування з ранжуванням

0-й цикл			1-й цикл			2-й цикл		
Масив чисел	Елементи маски	Масив рангів	Масив чисел	Елементи маски	Масив рангів	Масив чисел	Елементи маски	Масив рангів
2	1	1	1	1	1	0	0	1
5	1	1	4	1	1	3	1	2
3	1	1	2	1	1	1	1	2
4	1	1	3	1	1	2	1	2
3-й цикл			4-й цикл			5-й цикл		
Масив чисел	Елементи маски	Масив рангів	Масив чисел	Елементи маски	Масив рангів	Масив чисел	Елементи маски	Масив рангів
0	0	1	0	0	1	0	0	1
2	1	3	1	1	4	0	0	4
0	0	2	0	0	2	0	0	2
1	1	3	0	0	3	0	0	3

циклів, оскільки максимальним за значенням у масиві є число 5. Розмірність масиву чисел у цьому випадку не впливає на тривалість процесу, оскільки оброблення чисел виконується над усіма одночасно.

Особливість запропонованого позрізового сортування — це відсутність операції попарного порівнювання елементів масиву і складної мережі перекомутації елементів за результатами їхніх порівнянь. Крім того, тривалість процесу сортування не залежить від кількості n елементів сортованого масиву, а визначається величиною максимального елемента в масиві чисел, тобто залежить від тривалості операції декремента до повного обнулення всіх елементів масиву [22, 23]. Отже, максимальна тривалість процесу сортування в цьому випадку визначається величиною 2^n для двійково-кодованих чисел.

ЗАПИС ПАРАЛЕЛЬНОГО АЛГОРИТМУ ПОЗРІЗОВОГО СОРТУВАННЯ В БАЗИСІ САА ГЛУШКОВА

Для опису розглядуваного паралельного алгоритму сортування з урахуванням його особливостей потрібно розширити базис САА Глушкова [5, 9]. Це пов'язано з тим, що необхідно розглянути три масиви однакової довжини n :

- масив $M = a_1, a_2, \dots, a_n$ елементів, які сортуються;
- масив $R = r_1, r_2, \dots, r_n$ рангів відповідних елементів, які сортуються;
- масив $P = m_1, m_2, \dots, m_n$ елементів маски.

Введення масиву P потрібно для визначення за допомогою маскованого масиву рангів R місцезнаходження конкретного елемента з масиву M у відсортованому масиві.

Оскільки в процесі сортування задіяно одночасно всі елементи кожного масиву, то покажчики і маркери для їхніх розміток не застосовуються. На позначених масивах використовуються базисні умови (предикати) і оператори з орієнтацією на відомий базис регулярних схем (РС) для алгоритмів сортування в САА Глушкова [5], а саме:

- $Z(i)$ істинне, якщо виконано умову $\exists a_i = 0$;
- $Z(n)$ істинне, якщо виконано умову $\forall a_i = 0$;
- $p(i)$ істинне, якщо виконано умову $\exists m_i = 0$;

- $\text{OUT}(R)$ — оператор виведення результату;
- FIN — оператор завершення роботи РС.

У розглядуваному алгоритмі сортування зоною предикатів і операторів вважаються всі елементи трьох масивів, оскільки виконання операторів і перевірка істинності предикатів здійснюється одночасно над усіма елементами кожного масиву. Надалі для визначення базисних операторів, що вводяться для опису алгоритму сортування, потрібно скористатися основними операціями, що належать до сигнатури розширених САА-М [4, 5], а саме:

- диз'юнкція $\alpha \vee \beta$;
- композиція $A \times B$, тобто послідовне застосування операторів A і B ;
- альтернатива $[\alpha] (A \vee B)$, тобто якщо α істинне, то A , інакше B ;
- цикл $[\alpha] \{A\}$, тобто виконувати A поки α хибне, якщо α істинне — кінець циклу.

Крім того, можна скористатися позначенням $A + B$ для паралельного застосування операторів A і B [7]. Отже, для виконання операцій декремента, інкремента і маскування потрібно представити базисні оператори таким чином:

- встановлення паралельно початкових (одичних) станів n елементів масиву рангів R :

$$\text{SET}_n(\overline{r_1, r_n}) ::= (\overline{r_1, r_n} = 1),$$

- встановлення паралельно початкових (одичних) станів n елементів маски масиву P :

$$\text{SET}_n(\overline{m_1, m_n}) ::= (\overline{m_1, m_n} = 1).$$

Складові оператори в цьому випадку матимуть вигляд із застосуванням альтернативи:

- операція декремента над n елементами масиву M :

$$\text{DEC}_n(\overline{a_1, a_n}) ::= [a_i = 0] (a_i - 0 \vee a_i - 1)$$

або

$$\text{DEC}_n(\overline{a_1, a_n}) ::= [Z(i)] (a_i - 0 \vee a_i - 1); \quad (1)$$

- операція інкремента над n елементами масиву R :

$$\text{INC}_n(\overline{r_1, r_n}) ::= [m_i = 0] (r_i + 0 \vee r_i + 1)$$

або

$$\text{INC}_n(\overline{r_1, r_n}) ::= [p(i)] (r_i + 0 \vee r_i + 1); \quad (2)$$

- операція маскування над n елементами масиву P :

$$\text{MASK}_n(\overline{m_1, m_n}) ::= [a_i = 0] (m_i = 0 \vee m_i = 1)$$

або

$$\text{MASK}_n(\overline{m_1, m_n}) ::= [Z(i)] (m_i = 0 \vee m_i = 1). \quad (3)$$

В результаті паралельний алгоритм позрізового сортування з формуванням рангів у термінах САА Глушкова можна записати в такий спосіб:

$$\text{SORT}^n(\overline{a_1, a_n}) ::= \text{SET}_n(\overline{r_1, r_n}) + \text{SET}_n(\overline{m_1, m_n}) \times \\ \times [Z(n)] \{ \text{DEC}_n(\overline{a_1, a_n}) \times \text{MASK}_n(\overline{m_1, m_n}) \times \text{INC}_n(\overline{r_1, r_n}) \} \times \text{OUT}(R) \times \text{FIN}. \quad (4)$$

Більш детально з урахуванням опису складових операторів у вигляді (1)–(3) цей алгоритм сортування набуде вигляду

$$\begin{aligned} \text{SORT}^n(\overline{a_1, a_n}) &::= \text{SET}_n(\overline{r_1, r_n}) + \text{SET}_n(\overline{m_1, m_n}) \times \\ &\times [Z(n)] \{ [Z(i)](a_i - 0 \vee a_i - 1) \times [Z(i)](m_i = 0 \vee m_i = 1) \times \\ &\times [p(i)](r_i + 0 \vee r_i + 1) \} \text{OUT}(R) \times \text{FIN}. \end{aligned} \quad (5)$$

Аналізуючи записи (4) і (5), бачимо, що основна частина паралельного алгоритму позрізового оброблення — це цикл, який завершується після обнулення всіх елементів сортованого масиву. У самому циклі виконуються операції декременту і маскуванню рангів за цифровим значенням елементів маски з урахуванням обнулення конкретного елемента числового масиву, який сортується. Після кожного обнулення елемента сортованого масиву виконується операція інкременту рангів з урахуванням сформованих елементів маски.

ВИСНОВОК

Використовуючи апарат базисних предикатів і операцій, які складають сигнатуру розширених САА Глушкова, можна описати будь-який різновид послідовних і паралельних алгоритмів, що показово на прикладі алгоритмів сортування. При цьому характерною рисою базису САА Глушкова є його гнучкість і адаптованість, що підтверджує властиву йому функціональну потужність для різноманітних застосувань.

СПИСОК ЛІТЕРАТУРИ

1. Knuth D.E. The art of computer programming. Vol. 3, Sorting and searching. Reading: Addison-Wesley Longman, Inc., 1998. 800 p.
2. Pratt W.K. Introduction to digital image processing. Reading: Taylor and Francis Group, Inc., 2014. 371 p.
3. Осинский В.И., Мартынюк Т.Б., Козлов А.А., Мохамед Салем Нассер Мохамед. Особенности оптоэлектронной реализации сортирующей нейросети. *Опτικο-електронні інформаційно-енергетичні технології*. 2009. Т. 18, № 2. С. 58–67.
4. Цейтлин Г.Е. Проектирование последовательных алгоритмов сортировки: классификация, трансформация, синтез. *Программирование*. 1989. № 3. С. 3–24.
5. Цейтлин Г.Е. Распараллеливание алгоритмов сортировки. *Кибернетика*. 1989. Т. 24, № 6. С. 67–74.
6. Яценко Е.А. Регулярные схемы алгоритмов адресной сортировки и поиска. *Управляющие системы и машины*. 2004. № 5. С. 61–66.
7. Kozhemiako V.P., Martyniuk T.B., Khomyuk V.V. Distinctive features of structural programming of synchronous sorting algorithms. *Cybernetics and Systems Analysis*. 2006. Vol. 42, N 5. P. 714–723.
8. Цейтлин Г.Е. Структурное программирование задач символьной мультиобработки. - *Кибернетика*. 1983. Т. 19, № 5. С. 22–30.
9. Цейтлин Г.Е. Алгебры Глушкова и теория клонов. *Кибернетика и системный анализ*. 2003. Т. 39, № 4. С. 48–58.
10. Андон Ф.И., Дорошенко А.Е., Цейтлин Г.Е., Яценко Е.А. Алгеброалгоритмические модели и методы параллельного программирования. Киев: Академперіодика, 2007. 634 с.
11. Андон Ф.И., Дорошенко А.Е., Жереб К.А. Программирование высокопроизводительных параллельных вычислений: формальные модели и графические ускорители. *Кибернетика и системный анализ*. 2011. Т. 47, № 4. С. 176–187.

12. Андон Ф.И., Дорошенко А.Е., Бекетов А.Г., Иовчев В.А., Яценко Е.А. Инструментальные средства автоматизации параллельного программирования на основе алгебры алгоритмов. *Кибернетика и системный анализ*. 2015. Т. 51, № 1. С. 162–170.
13. Kohonen T. Content-addressable memories. Berlin; Heidelberg: Springer-Verlag, 1987. 388 p.
14. Thurber K.J. Large scale computer architecture: Parallel and associative processors. NJ.: Hayden Book Company, 1976. 324 p.
15. Martyniuk T.B. Structure of associative processor with bitwise serial processing of data. *Engineering Simulation*. 1997. Vol. 14. P. 383–389.
16. Martyniuk T., Vasilyeva T., Suprigan V., AL-Heyari M. Features of sorting memory realization. *Proceedings of SPIE (The International Society for Optical Engineering)*. 2001. Vol. 4425. P. 89–91.
17. Мартынюк Т., Куперштейн Л., Кожемяко А. Аспекты разностно-срезовой обработки данных в нейроструктурах. LAMBERT Academic Publishing RU, 2018. 60 p.
18. Martyniuk T.B., Khomyuk V.V. Data array multiprocessing by difference slices. *Cybernetics and Systems Analysis*. 2011. Vol. 47, N 6. P. 948–953. <https://doi.org/10.1007/s10559-011-9374-7>.
19. Martyniuk T.B., Kozhemiako A.V., Kupershtein L.M. Formalization of the object classification algorithm. *Cybernetics and Systems Analysis*. 2015. Vol. 51, N 5. P. 751–756. <https://doi.org/10.1007/s10559-015-9767-0>.
20. Lorin H. Sorting and sort systems. Mass.: Addison-Wesley Publishing Company, 1975. 373 p.
21. Кучеренко К.И., Очин Е.Ф. Процессоры двумерной медианной фильтрации изображений на основе сортирующих сетей. *Автоматика*. 1988. № 2. С. 13–19.
22. Мартинюк Т.Б., Денисюк Н.О., Круківський Б.І. Асоціативні процесори з паралельно-последовною обробкою даних. *Інформаційні технології та комп'ютерна інженерія*. 2019. Т. 44, № 1. С. 27–36. <https://doi.org/10.31649/1999-9941-2019-44-1-27-36>.
23. Мартинюк Т.Б., Круківський Б.І. Модель паралельного сортувальника масиву чисел. *Вісник Вінницького політехнічного інституту*. 2020. Т. 152, № 5. С. 49–55. <https://doi.org/10.31649/1997-9266-2020-152-5-49-55>.

T.B. Martyniuk, B.I. Krukivskyi

PECULIARITIES OF THE PARALLEL SORTING ALGORITHM WITH RANK FORMATION

Abstract. A new approach to parallel sorting of an array of numbers with the formation of their ranks is analyzed. In the process of sorting, such operations are implemented as a decrement operation for the processing of elements of a numerical array and an increment operation for the formation of their ranks. A description of the parallel sorting algorithm with the formation of ranks in the basis of Glushkov's System of Algorithmic Algebras (SAA) is proposed.

Keywords: system of algorithmic algebras, parallel sorting, numerical slice, rank, mask.

Надійшла до редакції 12.07.2021