УДК 519.681.5
doi.org/10.15407/fmmit2022.34-35.077

Паралельні методи та алгоритми розв’язання задач цифрової фільтрації масивів даних

Михайло Яджак
Доктор фізико-математичних наук, старший науковий співробітник, Інститут прикладних проблем механіки і математики ім. Я. С. Підстригача НАН України, вул. Наукова, 3-Б, м. Львів, 79060, Україна, Львівський національний університет імені Івана Франка, вул. Університетська, 1, м. Львів, 79000, Україна, e-mail: yadzhak_ms@ukr.net

Запропоновано квазісистолічний метод обчислень для розв’язання задач цифрової фільтрації різної розмірності на спеціалізованих обчислювальних засобах – квазісистолічних структурах. Цей метод дозволяє будувати оптимальні за швидкодією та використанням пам’яті паралельно-конвеєрні алгоритми обчислень. На основі ідей методу пірамід для розпаралелювання циклів побудовано паралельні алгоритми з автономними гілками для розв’язання задач фільтрації на кластерах, гібридних архітектурах та комп’ютерах з багатоядерним процесором. Одержано теоретичні оцінки прискорення, що підтверджують високу ефективність побудованих паралельних алгоритмів фільтрації. Для окремих алгоритмів з автономними гілками одержано реальні оцінки прискорення, які добре узгоджуються з теоретичними. Одержані результати можна використати для попереднього опрацювання великих масивів вхідних даних у різних предметних галузях.

Ключові слова: цифрова фільтрація, квазісистолічний метод, паралельно-конвеєрний алгоритм, метод пірамід, розпаралелювання циклів, прискорення обчислень, автономні гілки, кластер.

Вступ. Під час дослідження складних об’єктів та процесів [1], зокрема на транспорті, в зв’язку, торгівлі, освіті, у банківській справі, використовують значні обсяги вхідних даних, які надходять неперервно. Зазвичай ці дані є неточними, спотвореними або пошкодженими. Тому для подальшого використання їх потрібно попередньо опрацювати, тобто відфільтрувати, і у більшості випадків – в режимі реального часу. З цією метою було запропоновано низку паралельних алгоритмів цифрової фільтрації, орієнтованих на різні типи архітектур обчислювальних систем [2–4]. Однак, згадані алгоритми мали певні недоліки, пов’язані з затримками потоків даних, зменшенням швидкості збіжності обчислювального процесу, недостатнім залученням резервів розпаралелювання, строгим обмеженням галузі застосування тощо. Тому актуальною залишалася розробка для розв’язання задач цифрової фільтрації (ЗЦФ) різної розмірності високопаралельних алгоритмів, в яких ці недоліки усунуті частково або повністю.

1. Формулювання проблеми.

У загальному випадку розглядувана ЗЦФ полягає у виконанні C переобчислень згладжування масиву значень N змінних через рухоме вікно розміром M [5].

Наприклад, у двовимірному випадку переобчислення згладжування значень $x_{i_1,i_2}\ (i_1=\overline{1,l_1};\ i_2=\overline{1,l_2})$ здійснюються за формулою

$$x_{i_1,\,i_2} = \sum_{s_1=-m_1}^{m_1} \quad \sum_{s_2=-m_2}^{m_2} x_{i_1+s_1,\,i_2+s_2}\, f_{s_1,\,s_2}\,. \tag{1}$$

Тут $N = l_1 l_2$, $M = (2m_1+1)(2m_2+1)$, а вагові коефіцієнти $f_{s_1,s_2}\,(s_1 = \overline{-m_1, m_1};\ s_2 = \overline{-m_2, m_2})$ та «замежові» значення $x_{1-m_1,\,j^*}, x_{2-m_1,\,j^*}, \ldots, x_{0,\,j^*}$; $x_{l_1+1,\,j^*}, x_{l_1+2,\,j^*}, \ldots, x_{l_1+m_1,\,j^*}$ $(j^* = \overline{1-m_2,\, l_2+m_2})$; $x_{i^*,\,1-m_2}, x_{i^*,\,2-m_2}, \ldots, x_{i^*,\,0}$; $x_{i^*,\,l_2+1}, x_{i^*,\,l_2+2}, \ldots, x_{i^*,\,l_2+m_2}$ $(i^* = \overline{1, l_1})$ є відомими константами. Зазвичай для більшості практичних задач виконуються нерівності $M << N,\ C << N$.

Звичайний послідовний алгоритм розв'язання двовимірної ЗЦФ на підставі використання (1) має вигляд [6]:

$$\begin{array}{l}
FOR\quad t = 1,\, C \quad DO \\
\{\ FOR\quad i_1 = 1,\, l_1 \quad DO \\
\{\ FOR\quad i_2 = 1,\, l_2 \quad DO \\
\{\ p_1 = 0 \\
FOR\quad s_1 = -m_1,\, m_1 \quad DO \\
FOR\quad s_2 = -m_2,\, m_2 \quad DO \\
\{\ p_1 = p_1 + x_{i_1+s_1,\, i_2+s_2} * f_{s_1,s_2}\ \} \\
x_{i_1,\, i_2} = p_1\ \}\ \}\ \}.
\end{array} \tag{2}$$

Характерною особливістю алгоритму (2) є те, що для переобчислення значення змінної x_{i_1,i_2} на t-му кроці використовуються значення, які вже є переобчисленими на цьому ж кроці.

Інший послідовний алгоритм розв'язання двовимірної ЗЦФ виглядає так [7]:

$$\begin{array}{l}
FOR\quad t = 1,\, C \quad DO \\
\{\ FOR\quad i_1 = 1,\, l_1 \quad DO \\
\{\ FOR\quad i_2 = 1,\, l_2 \quad DO \\
\{\ p = 0 \\
FOR\quad s_1 = -m_1,\, m_1 \quad DO \\
\{\ FOR\quad s_2 = -m_2,\, m_2 \quad DO \\
\{\ p = p + x^{t-1}_{i_1+s_1,\, i_2+s_2} * f_{s_1,s_2}\ \}\ \} \\
x^{t}_{i_1,\, i_2} = p\ \}\ \}\ \}.
\end{array} \tag{3}$$

У цьому алгоритмі $x^0_{i_1,\,i_2}$, $x^t_{i_1,i_2}$ – відповідно початкове значення змінної x_{i_1,i_2} та значення цієї ж змінної, переобчислене на t-му кроці.

У наведеній конструкції (3) для переобчислення значення x_{i_1,i_2} на t-му кроці використовуються значення, переобчислені виключно на (t-1)-му кроці.

Вирішувана в роботі проблема полягає в розробці методів і високопаралельних алгоритмів цифрової фільтрації, еквівалентних за інформаційним графом до відповідних послідовних алгоритмів (у випадку розв'язання двовимірної ЗЦФ – це алгоритми (2) та (3)).

2. Квазісистолічний метод.

Нами запропоновано квазісистолічний метод організації обчислень для розв'язання задач фільтрації різної розмірності [5, 6], який полягає в наступному:

- значення N змінних переобчислюються в окремих, зсунутих між собою гілках, тобто головним прийомом підвищення міри паралелізму тут, крім власне розпаралелювання, є конвеєризація;
- дозволяється одночасне пересилання даних із однієї інстанції одразу в M «точок прийому»;
- для переобчислення значення довільної змінної на деякому кроці використовується максимальна кількість уже переобчислених на ньому значень;
- функціональна еквівалентність з послідовним способом обчислень забезпечується виконанням операцій додавання в строго визначеному порядку [5];
- кожне з переобчислених значень використовується в (M-1) гілці, де воно є аргументом, на наступному ж такті.

На основі описаного методу побудовано паралельно-конвеєрні алгоритми (ПКА) розв'язання ЗЦФ різної розмірності. Зокрема, такий алгоритм для розв'язання двовимірної задачі фільтрації має вигляд:

$$
\begin{array}{l}
FOR \quad i_1 = 1,\, l_1 \quad DO \quad SYNCH \\
\{ \; DELAY \; ((4m_2+2)(i_1-1)) \\
FOR \quad i_2 = 1,\, l_2 \quad DO \quad SYNCH \\
\{ \; DELAY \; (2i_2-2) \\
FOR \quad t = 1,\, C \quad DO \\
BEGIN \\
FOR \quad u = 1,\, (2m_1+1)(2m_2+1) \quad DO \\
BEGIN \\
x_{i_1,i_2} = IF \; (u=1) \;\; THEN \;\; f_{0,\,0} * x_{i_1,i_2} \;\; ELSE \;\; x_{i_1,\,i_2} + y_{i_1,\,i_2}, \\
y_{i_1,\,i_2} = f_{h_1(u,m_1,m_2),\,h_2(u,m_2)} * x_{i_1+h_1(u,m_1,m_2),\,i_2+h_2(u,m_2)} \\
END \\
DELAY \; (2) \\
END \; \} \}.
\end{array}
\tag{4}
$$

У наведеному алгоритмі *SYNCH* – це тип паралельності [8], що задає синхронізацію між паралельними гілками після виконання в них кожної операції; *DELAY*(p) – оператор, який здійснює затримку на p тактів; передбачається, що оператори $x_{i_1,i_2} = \ldots$ і $y_{i_1,i_2} = \ldots$, розділені комою, виконуються синхронно. Функції $h_1(u,\, m_1,\, m_2)$, $h_2(u,\, m_2)$ – це відповідно такі умовні оператори:

$$IF \; ((u/2) = [u/2]) \; THEN \; [(u-1)/(4m_2+2)] - m_1 \; ELSE$$

$$[(u+2m_2+2)/(4m_2+2)],$$
$$IF\ ((u/2)=[u/2])\ \ THEN\ \ u/2-m_2-1-(2m_2+1)[(u-1)/(4m_2+2)]$$
$$ELSE\ \ u/2+1/2-(2m_2+1)[(u+2m_2+2)/(4m_2+2)].$$

Тут запис $[a]$ означає цілу частину числа a.

Доведено оптимальність [6] алгоритму (4) за швидкодією та використанням пам’яті у вказаному класі алгоритмів, які є еквівалентними за інформаційним графом з точністю до виконання співвідношень асоціативності та комутативності. Побудований ПКА зорієнтований на реалізацію на відповідній спеціалізованій обчислювальній системі – квазісистолічній структурі [9], яку можна розглядати, як окремий обчислювальний вузол кластера для ефективного виконання спеціальних обчислень.

Необхідно зазначити, що алгоритми (2) та (4) є еквівалентними за інформаційним графом. Якщо припустити, що час виконання операції додавання та множення є однаковим і дорівнює одному такту, то отримуємо, що послідовний алгоритм (2) потребує виконання

$$2(2m_1+1)(2m_2+1)Cl_1l_2$$

тактів, а ПКА (4) потребує

$$((2m_1+1)(2m_2+1)+2)C+(4m_2+2)(l_1-1)+2l_2-4$$

такти. На основі наведених оцінок складності алгоритмів одержуємо формулу для прискорення (4):

$$S_1=\frac{2(2m_1+1)(2m_2+1)Cl_1l_2}{((2m_1+1)(2m_2+1)+2)C+(4m_2+2)(l_1-1)+2l_2-4}.$$

Мінімальний розмір рухомого вікна маємо у разі $m_1=m_2=1$. Унаслідок тривіального обчислення для $C=10,\ l_1=100,\ l_2=200$ одержуємо, що $S_1\approx 3272$. У випадку $C=50,\ l_1=1000,\ l_2=2000$ одержуємо $S_1\approx 170700$.

Якщо припустити, що максимальний розмір видимого вікна, враховуючи площу робочого екрану, одержується для $m_1=10,\ m_2=40$, то у разі $C=10,$ $l_1=100,\ l_2=200$ встановлюємо оцінку прискорення: $S_1\approx 20300$. Для $C=50$, $l_1=1000,\ l_2=2000$ одержуємо $S_1\approx 13\cdot 10^5$. Звідси випливає, що прискорення ПКА (4) є суттєвим.

Запропонований квазісистолічний метод обчислень розвинуто для побудови оптимальних за швидкодією ПКА розв’язання задач каскадної цифрової фільтрації різної розмірності та задачі фільтрації з використанням процедури адаптивного згладжування [10].

2. Застосування методу пірамід для розпаралелювання циклів.

Для розв’язання ЗЦФ різної розмірності розглянуто послідовні алгоритми, в яких для переобчислення масиву значень змінних на заданому кроці беруться значення, переобчислені на попередньому кроці. Застосовуючи до цих алгоритмів метод пірамід [8, 11] для розпаралелювання циклів, одержуємо паралельні алгоритми з автономними гілками [7]. Сам метод пірамід полягає у відшуканні результуючих ітерацій в просторі ітерацій вихідного циклу та формуванні для кожної з них підмножини ітерацій, які на неї впливають. На основі кожної із цих підмножин будується паралельна автономна гілка. Геометрично результуюча ітерація та відповідна їй підмножина утворюють деяку піраміду. Зокрема, застосовуючи метод пірамід до алгоритму (3), одержуємо такий паралельний алгоритм:

$$
\begin{array}{l}
FOR\quad ALL\quad (k_1,k_2)\in\{(k_1,k_2):\ k_1=\overline{1,l_1};\ k_2=\overline{1,l_2}\}\quad DO\quad AUTON\\
\{\ FOR\quad t=1,\ C\quad DO\\
\{\ FOR\quad i_1=\max\{1,\ (t-C)m_1+k_1\},\ \min\{l_1,\ (C-t)m_1+k_1\}\quad DO\\
\{\ FOR\quad i_2=\max\{1,\ (t-C)m_2+k_2\},\ \min\{l_2,\ (C-t)m_2+k_2\}\quad DO\\
\{\ p=0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (5)\\
FOR\quad s_1=-m_1,\ m_1\quad DO\\
\{\ FOR\quad s_2=-m_2,\ m_2\quad DO\\
\{\ p=p+x^{t-1}_{i_1+s_1,\ i_2+s_2}*f_{s_1,s_2}\ \}\ \}\\
x^t_{i_1,\ i_2}=p\ \}\ \}\ \}\ \}.
\end{array}
$$

У наведеному алгоритмі *AUTON* є типом паралельності [8], який вказує на те, що між паралельними гілками немає обмінів інформацією. Конструкція (5) задає паралельне виконання l_1l_2 автономних гілок. За зроблених вище припущень стосовно тривалості операцій цей алгоритм для свого виконання потребує

$$2(2m_1+1)(2m_2+1)C(1+((2/3)m_1m_2(2C-1)+m_1+m_2)(C-1))$$

тактів. Послідовний алгоритм (3), який є еквівалентним до (5) за інформаційним графом, вимагає

$$2(2m_1+1)(2m_2+1)Cl_1l_2$$

тактів.

Прискорення паралельного алгоритму (5) можна обчислити за формулою

$$S_2=l_1l_2/(1+(2m_1m_2(2C-1)/3+m_1+m_2)(C-1)).$$

Зазвичай на практиці $l_1>>m_1,\ l_2>>m_2,\ l_1>>C,\ l_2>>C$ та $l_1,\ l_2$ відрізняються від m_1C^2, m_2C^2 ($l_1>m_1C^2,\ l_2>m_2C^2$) відповідно не менш, як на декілька порядків, тому S_2 буде суттєвим. Наприклад, для параметрів задачі фільтрації $l_1=100,\ l_2=200,\ m_1=m_2=1,\ C=5$ отримуємо $S_2\approx 606$, а для $l_1=l_2=1000$, $m_1=m_2=5,\ C=10$ маємо: $S_2\approx 340$.

Нами запропоновано деякий підхід до покращення процедури згладжування в побудованих алгоритмах з автономними гілками. Зокрема, у (5) процес згладжування можна дещо покращити, якщо останні п'ять рядків замінити наступним фрагментом:

$$
\begin{array}{l}
\{\ p_{i_1,i_2}=0\\
FOR\quad s_1=-m_1,\ m_1\quad DO\\
\{\ FOR\quad s_2=-m_2,\ m_2\quad DO
\end{array}
$$

$\{\ p_{i_1,i_2} = p_{i_1,i_2} + x_{i_1+s_1,\, i_2+s_2} * f_{s_1,s_2}\ \}\ \}$

$x_{i_1,\, i_2} = p_{i_1,i_2}\ \}\ \}\ \}\ \}.$

Одержана унаслідок такої заміни паралельна конструкція дозволить у кожній гілці під час переобчислення значення деякої змінної на заданому кроці використовувати певну кількість значень, які є вже переобчисленими на цьому ж кроці.

Побудовані паралельні алгоритми фільтрації з автономними гілками зорієнтовані на реалізацію на сучасних високопродуктивних засобах – кластерах, гібридних архітектурах та комп'ютерах з багатоядерним процесором. На основі цих алгоритмів розроблено відповідні алгоритми з обмеженим паралелізмом, які враховують обсяг реально доступних обчислювальних ресурсів (кількість процесорів (ядер), обчислювальних вузлів; обсяг використовуваної пам'яті, продуктивність комунікаційного середовища тощо).

Проведено низку чисельних експериментів на комп'ютері з багатоядерним процесором для отримання реального прискорення розроблених алгоритмів фільтрації з автономними гілками. Наприклад, прискорення паралельного алгоритму розв'язання двовимірної ЗЦФ порівняно з послідовною реалізацією із залученням чотирьох та восьми ядер становило відповідно 2,80–3,62 та 3,10–7,50.

Висновки. Запропоновано паралельні методи і ефективні паралельно-конвеєрні алгоритми та паралельні алгоритми з автономними гілками для розв'язання задач фільтрації різної розмірності відповідно на перспективних спеціалізованих засобах – квазісистолічних структурах та сучасних засобах універсального призначення – кластерах, гібридних архітектурах [12] та комп'ютерах з багатоядерним процесором. Одержано теоретичні та деякі реальні оцінки прискорення для запропонованих паралельних алгоритмів фільтрації. Розроблені паралельні методи та алгоритми обчислень можна використати для попереднього опрацювання в режимі реального часу великих масивів вхідних даних у різноманітних предметних галузях із застосуванням сучасних програмних та апаратних засобів. Подальші дослідження автор вбачає у більш ширшому використанні запропонованих методів організації паралельних обчислень для розв'язання задач фільтрації більшої розмірності на підставі адаптивного згладжування.

Література

1. Поліщук О. Д., Яджак М. С. Мережеві структури та системи: IV. Паралельне опрацювання результатів неперервного моніторингу. *Системні дослідження та інформаційні технології*. 2019. № 2. С. 105–114.
2. Каневский Ю. С. Систолические процессоры. Киев: Тэхника, 1991. 173 с.
3. Valkovskii V. A. An optimal algorithm for solving the problem of digital filtering. *Pattern Recognition and Image Analysis*. 1994. Vol. 4, N 3. P. 241–247.
4. Тимченко О. В. Різницеві методи цифрової фільтрації. Львів: Фенікс, 1999. 388 с.
5. Anisimov A. V. and Yadzhak M. S. Construction of optimal algorithms for mass computations in digital filtering problems. *Cybernetics and Systems Analysis*. 2008. Vol. 44, N 4. P. 465–476.
6. *Яджак М. С.* Високопаралельні алгоритми та засоби для розв'язання задач масових арифметичних і логічних обчислень. *Автореф. дис. ... д. ф.-м. н.: спеціальність 01.05.03 – математичне та програмне забезпечення обчислювальних машин і систем*. Київ: КНУ імені Тараса Шевченка, 2009. 33 с.

7. *Яджак М. С.* Деякі паралельні алгоритми розв'язання задач цифрової фільтрації. *Матеріали VIII міжнар. наук.-практ. конф. «Математика в сучасному технічному університеті», Київ, 27–28 грудня 2019 р.* Вінниця: Видавець ФОП Кушнір Ю. В., 2020. С. 172–176.
8. *Вальковский В. А.* Распараллеливание алгоритмов и программ. Структурный подход. М.: Радио и связь, 1989. 176 с.
9. *Яджак М. С.*, Тютюнник М. І., Бекас Б. О. Апаратні засоби реалізації паралельно-конвеєрних алгоритмів цифрової фільтрації з використанням адаптивного згладжування. *Науковий вісник НЛТУ України.* 2014. Вип. 24.6. С. 335–344.
10. *Тютюнник М. І.* Паралельні алгоритми комплексного оцінювання стану та якості функціонування складних систем. *Автореф. дис. ... к. т. н.: спеціальність 01.05.03 – математичне та програмне забезпечення обчислювальних машин і систем*. Київ: КНУ імені Тараса Шевченка, 2020. 21 с.
11. *Вальковський В. О., Яджак М. С.* Проблеми подальшого розвитку та модифікації методу пірамід для розпаралелювання циклів. *Математичні методи та фізико-механічні поля*. 2000. **43**, № 1. С. 68–75.
12. *Попов О. В., Рудич О. В.* До розв'язування систем лінійних рівнянь на комп'ютерах гібридної архітектури. *Математичне та комп'ютерне моделювання.* Серія: Фізико-математичні науки: зб. наук. праць. 2017. Вип. 15. С. 158–164.

Parallel methods and algorithms for solving problems of digital filtering of data arrays

Yadzhak Mykhailo

A quasisystolic computation method for solving digital filtering problems of various dimensions on specialized computing means – quasisystolic structures is proposed. This method makes it possible to build parallel-pipeline computation algorithms that are optimal in speed and memory usage. Based on the ideas of the pyramid method for parallelization of cycles, parallel algorithms with autonomous branches were built for solving filtering problems on clusters, hybrid architectures and computers with a multi-core processor. Theoretical speed-up estimates were obtained, which confirm the high efficiency of the constructed parallel filtering algorithms. For individual algorithms with autonomous branches, real speed-up estimates were obtained, which are in good agreement with the theoretical ones. The obtained results can be used for preliminary processing of large arrays of input data in various subject areas.

Отримано 03.10.22.