



УДК 511.222:001.103

РЕАЛІЗАЦІЯ ДЕЯКИХ ПАРАЛЕЛЬНИХ АЛГОРИТМІВ ЦИФРОВОЇ ФІЛЬТРАЦІЇ МАСИВІВ ДАНИХ

Ростислав ВДОВИЧ¹, Михайло ЯДЖАК^{1, 2}

*¹Інститут прикладних проблем механіки і математики
імені Я. С. Підстригача НАН України, м. Львів, Україна;*

*²Львівський національний університет
імені Івана Франка, м. Львів, Україна*

Вступ. Під час дослідження об'єктів складних ієрархічно-мережевих систем зазвичай використовують значні обсяги вхідних даних про них (характеристики, особливості режимів функціонування, можливі критерії оцінювання тощо) [1, 2]. Перед подальшим використанням ці дані треба попередньо опрацювати, використовуючи процедуру цифрової фільтрації. Здебільшого її необхідно виконувати в режимі реального часу. Для цього запропоновано низку паралельних алгоритмів фільтрації, орієнтованих на різні типи архітектур паралельних обчислювальних засобів (систолічні та квазісистолічні структури, комп'ютери зі структурно-процедурною організацією обчислень, кластери, гібридні архітектури, комп'ютери з багатоядерними процесорами тощо) [3-5]. Зокрема, у праці [5] проаналізовано паралельні алгоритми фільтрації, що реалізують синхронну схему

обчислень, і на їх основі побудовано алгоритми з автономними гілками й обмеженим паралелізмом. Утім, не достатньо дослідженою є проблема аналізу побудованих алгоритмів з використанням різних програмних засобів реалізації паралелізму. Цій проблемі і присвячено нашу працю. Далі коротко опишемо паралельні алгоритми фільтрації, для яких оцінюватимемо реальне прискорення.

Паралельні алгоритми цифрової фільтрації. Розглянемо задачу цифрової фільтрації (ЗЦФ), що загалом полягає у виконанні S переобчислень згладжування масиву значень N змінних через рухоме вікно розміром M [4].

Для розв'язання сформульованої одновимірної ЗЦФ використовуємо паралельний алгоритм (A1), який реалізує синхронну схему обчислень. Згідно з цим алгоритмом, для переобчислення значень змінних на j -му кроці беруться значення, переобчислені тільки на $(j-1)$ -му кроці. Зауважимо, що на кожному кроці переобчислень здійснюється синхронізація паралельних гілок.

У [5] запропоновано паралельний алгоритм (A2) фільтрації з автономними N гілками. Результатом його роботи є переобчислені значення N змінних на S -му кроці. Кожна паралельна гілка алгоритму складається з ітерацій, які впливають на результуючу. Однак, в A2 спостерігається дублювання обчислень на низці ітерацій, що робить його неекономним. Попри це, здебільшого, урахувавши, що найчастіше на практиці $N \gg M$ та $N \gg S$, цей алгоритм доволі ефективний.

Дещо зменшити частку дублювань обчислень у паралельних гілках дає змогу використання алгоритму (A3) фільтрації з обмеженим паралелізмом, який урахує, що $N > P$, де P – кількість реально виконуваних паралельних гілок [5]. Гілки в A3 суттєво завантажені корисною (обчислювальною) роботою. Цей алгоритм орієнтований на заданий обсяг доступних ресурсів (кількість процесорів, ядер, обчислювальних вузлів) у наявному паралельному комп'ютері.

Аналіз реального прискорення паралельних алгоритмів. Для аналізу вибрано алгоритми A1 та A3 і реалізовано їх на комп'ютері з шестиядерним процесором Intel Core i5-9600 KF на мові високого рівня C# у середовищі Visual Studio 2022 із залученням бібліотек System та System.Threading. Кожен із цих паралельних алгоритмів фільтрації був виконаний двома способами із застосуванням ThreadPool та Parallel.For. Чисельний експеримент проведено для

різних наборів значень параметрів задачі фільтрації. Для кожного з цих наборів використано по декілька масивів вхідних даних.

Час виконання паралельних алгоритмів А1 та А3 ми порівнювали з часом виконання відповідного послідовного алгоритму, еквівалентного до них за інформаційним графом [3, 4]. Найбільше прискорення А1 з використанням ThreadPool становило 3,47 раза, а з використанням Parallel.For – 4,18 раза. Для алгоритму А3, використовуючи згадані способи реалізації паралелізму, відповідно, ми отримали найбільше прискорення в 5,08 та 5,05 раза.

Висновок. Реалізовано паралельний алгоритм, що використовує синхронну схему обчислень, та алгоритм з обмеженим паралелізмом розв'язання одновимірної ЗЦФ на комп'ютері з багатоядерним процесором із застосуванням сучасних програмних засобів. Одержано реальне прискорення, що підтверджує високу ефективність цих алгоритмів. Результати роботи можуть бути використані для попереднього опрацювання в режимі реального часу великих масивів вхідних даних у різних предметних галузях, зокрема під час моделювання об'єктів і процесів у біологічних та природних системах, а також у фізичному вихованні та спорті.

Список використаних джерел

1. Polishchuk O. Big Data Processing in Complex Hierarchical Network Systems I: Structures and Information Flows / O. Polishchuk, D. Polishchuk, M. Tyutyunnyk, M. Yadzhak // AASCIT Communications. – 2016. – Vol. 3, No 3. – P. 112–118.
2. Поліщук О. Д. Моделі та методи комплексного дослідження складних мережевих систем та міжсистемних взаємодій / О. Д. Поліщук, М. С. Яджак. – Львів: Інститут прикладних проблем механіки і математики ім. Я. С. Підстригача НАН України, 2023. – 385 с.
3. Jadzhak M. S. On Optimal in One Class Algorithm for Solving Three-Dimensional Digital Filtering Problem / M. S. Jadzhak // Journal of automation and information sciences. – 2001. – Vol. 33, No 1. – P. 51–63.
4. Яджак М. С. Високопаралельні алгоритми та засоби для розв'язання задач масових арифметичних і логічних обчислень: автореф. дис. д. ф.- м. наук: [спец.] 01.05.03 «Математичне та програмне забезпечення обчислювальних машин і систем» / М. С. Яджак. – К.: КНУ ім. Тараса Шевченка, 2009. – 33 с.
5. Яджак М. С. Паралельні алгоритми цифрової фільтрації даних / М. С. Яджак // Кібернетика та системний аналіз. – 2023. – 59, № 1. – С. 46–56.