

ПОРІВНЯЛЬНИЙ АНАЛІЗ ТЕХНОЛОГІЙ ПАРАЛЕЛЬНОГО ПРОГРАМУВАННЯ НА ПРИКЛАДІ РОЗВ'ЯЗАННЯ СИСТЕМ ЛІНІЙНИХ РІВНЯНЬ

Барабан Сергій, Барабан Марія

Вінницький національний технічний університет

Анотація

В роботі проведено аналіз відомих точних методів розв'язування СЛАР, а також обрано метод Джордана-Гауса на основі якого розроблено схему розпаралелювання сегментів. Здійснено реалізації консольних додатків на основі технологій паралельного програмування OpenMP та MPI. Виконано дослідження швидкодії імплементованих додатків шляхом порівняння з послідовною реалізацією.

Abstract

The analysis of known exact methods for solving systems of linear algebraic equations is performed, as well as the Gauss–Jordan elimination based on which the scheme of segment parallelization is developed. Console applications based on OpenMP and MPI concurrent programming technologies are implemented. The performance of the implemented applications by comparison with the sequential implementation was performed.

Вступ

Задачі математичного моделювання, з формального погляду, не викликають труднощів, оскільки процес розв'язування систем рівнянь математичних моделей часто зводиться до розв'язування СЛАР, які виконують методом Гауса чи його модифікаціями. Однак під час розв'язування СЛАР високих порядків обчислювальна складність цих методів значна. Тому виникає необхідність застосування паралельних обчислень, що дають змогу досягти суттєвого підвищення продуктивності обрахунків.

Актуальність даної роботи полягає в тому, що використання новітніх методів реалізації паралельних алгоритмів (з можливістю інтеграції обчислень на графічних картах) [1] дозволить значно підвищити ефективність сучасних алгоритмів розв'язування СЛАР.

Метою роботи є дослідження сучасних технологій паралельних обчислень, проектування та імплементация прикладної програми для паралельних обрахунків, порівняння характеристик реалізованого паралельного алгоритму з відомим послідовним.

Розробка паралельного алгоритму

Відомі два основні класи розв'язування СЛАР: точні та ітераційні (наближені) методи [2]. Розв'язуючи СЛАР ітераційними методами, крім похибок округлення, слід враховувати також і похибку методу [3]. Розглянуто тільки алгоритми точних методів, так як саме один з них буде використовуватись для створення паралельного алгоритму і його реалізації мовою C++. Це пов'язано з неоднозначною придатністю наближених алгоритмів до розпаралелювання. Другорядним критерієм для вибору точного методу є необхідність мінімізації похибки програмних обрахунків [2]. Незважаючи на перевагу паралельної реалізації методу Холецкого над методом Гауса (в плані обчислень та економії пам'яті), за мету ставиться реалізація такого алгоритму, який дозволяє знаходити розв'язок різнотипних систем лінійних рівнянь (метод Холецкого доцільний для симетричних матриць), тому обрано метод Жордана-Гауса [4].

Враховуючи опис паралелізму для методу Гауса, можна прийти висновку, що піддаються розподілу на потоки такі сегменти алгоритму:

1. Ділення рядків на елемент, що лежить на головній діагоналі матриці.

2. Прямий хід Жордана-Гауса (занулення всіх елементів під головною діагоналлю).
3. Зворотний хід Жордана-Гауса (занулення всіх елементів над головною діагоналлю).
4. Множення оберненої матриці, отриманої після проведення зворотного ходу, на стовпець вільних членів.

Для оцінювання швидкодії паралельного алгоритму скористаємось формулою, за якою розраховується коефіцієнт прискорення: $K_n = T_{пс} / T_{пр}$, де $T_{пс}$ – час виконання послідовних обчислень, $T_{пр}$ – час виконання паралельних обчислень (при однакових розмірностях вхідних даних). На рисунку 1 показано графіки залежності часу виконання від вхідних даних.

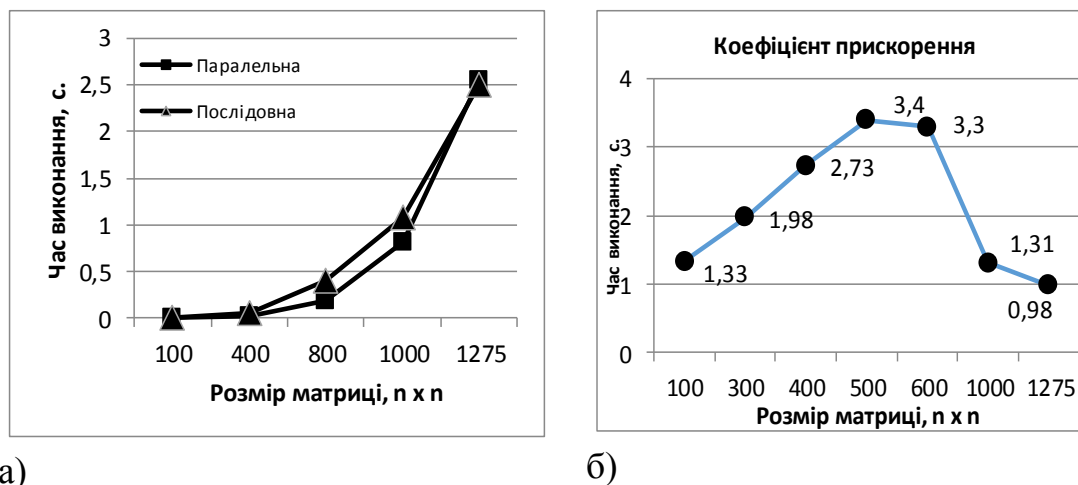


Рисунок 1 – Порівняння часу виконання паралельної та послідовної програм (а), зміна значення коефіцієнта прискорення зі збільшенням розмірності матриці (б)

З діаграми рисунка 1 (а) видно, що час виконання паралельних обчислень практично збігається з часом послідовних обчислень при $n \approx 1275$. При $n > 1275$ час виконання паралельних обчислень перевищує послідовні. Наприклад, при обрахунку матриці розмірністю 2500×2500 послідовний алгоритм справляється за $T_{пс} \approx 20$ с, а паралельний – за $T_{пр} \approx 25$ с. З діаграми видно, що максимальне прискорення паралельної програми $K_{n \max} = 3,4$. Можна зробити висновок, що на даному ПК найдоцільніше використовувати OpenMP-реалізацію при діапазоні $300 < n < 800$.

Список використаних джерел

1. Information technology in creating intelligent chatbots [Text] / Yarovy A.A., Kudriavtsev D.V., Varaban S.V., etc. // Proc. SPIE 11176, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019, 1117627 (6 November 2019); <https://doi.org/10.1117/12.2537415>
2. Харів А., Хвищун І. Паралельні методи розв'язання СЛАР з використанням архітектури CUDA / А. Харів, І. Хвищун // ISSN 2224-087X. Електроніка та інформаційні технології. – 2015. – №5. – с. 192-199.
3. Хвищун І. О. Програмування і математичне моделювання / І. О. Хвищун. – К. : ІнЮре, 2007.
4. Розв'язок систем лінійних рівнянь методом Жордана-Гаусса [Електронний ресурс] // Сайт для студентів спеціальності «Інформатика» – Режим доступу до ресурсу: <http://www.mathros.net.ua/metod-zhordana-gaussa-rozvjazok-system-linijnyh-rivnjan-metodom-zhordana-gaussa.html>.