

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Кафедра Системотехніки

Звіт

з практичної роботи №2

з дисципліни: «Технології Високопродуктивних Обчислень»

з теми: «Рішення найпростіших математичних завдань за допомогою алгоритмів
паралельних обчислень»

Виконав:

здобувач освіти першого
(бакалаврського) рівня освіти гр.

КНТ-22-1

Орлов О. С.

Варіант: №9

Перевірив:

Професор кафедри СТ
Міщеряков Ю. В.

2 РІШЕННЯ НАЙПРОСТІШИХ МАТЕМАТИЧНИХ ЗАВДАНЬ ЗА ДОПОМОГОЮ АЛГОРИТМІВ ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ

2.1 Мета роботи

Метою роботи є оволодіння основними навичками аналізу трудомісткості паралельних обчислень на прикладі паралельного алгоритму множення матриць, визначення основних характеристик трудомісткості та їх залежності від схеми обчислень та кількості процесорів. Отримання досвіду оцінки трудомісткості паралельних обчислень за допомогою основних теорем.

2.2 Хід роботи

Темою на розгляд у даній роботі є множення матриць. Елементи результуючої матриці розраховуються за наступною формулою: $c_{ij} = \sum_{k=1}^n a_{ik} * b(kj)$ Візьмемо дві матриці розмірністю $n \times n$. Для розрахунку одного елементу результуючої матриці знадобиться $n + (n - 1) = 2n - 1$ операцій (n множень, $n - 1$ додавань). Враховуючи n^2 елементів у вихідних матрицях, для розрахунку результату на одному процесорі (враховуючи, що одна операція займає одну одиницю часу) знадобиться рівно $(2n - 1) * n^2$ одиниць часу. Іншими словами, час розрахунку на одному процесорі можна описати так:

$$T_1 = (2n - 1) * n^2 \quad (2.1)$$

Спершу, розрахуємо всі основні показники для $p = n^2$. Враховуючи, що кількість процесорів дорівнює кількості елементів, можемо зазначити, що час на виконання у багатопоточному режимі буде дорівнювати часу обробки одного елемента $T_p = 2n - 1$.

$$S_p = \frac{T_1}{T_{n^2}} = \frac{(2n - 1) \cdot n^2}{2n - 1} = n^2 \quad (2.2)$$

$$E_p = \frac{S_p}{p} = \frac{n^2}{n^2} = 1 \quad (2.3)$$

$$C_p = T_p \cdot p = (2n - 1) \cdot n^2 \quad (2.4)$$

Відмітимо, що ефективність E_p дорівнює одиниці, що означає 100% використання всіх процесорів під час роботи алгоритму. Також, ціна дорівнює часу виконання на одному процесорі, що дає нам привід назвати цей розподіл роботи вартісно-оптимальним.

Наступним кроком, визначимо кількість процесорів і розподіл робіт, що забезпечує підвищення прискорення S_p відносно попереднього варіанту. Враховуючи, що ми не можемо впливати на кількість обробки елементів одним процесором, ми зменшимо кількість обробки p процесорами, взявши $p = n^3$. Ця міра дозволить нам зменшити час обробки всіх елементів до наступного:

$$T_p = T_{\text{mult}} + T_{\text{sum}} = 1 + \log_2 n \quad (2.5)$$

Де $T_{\text{mult}} = 1$ тому що ми розділили час множення елементів n на кількість процесорів n , та $T_{\text{sum}} = \log_2 n$, використовуючи каскадний алгоритм додавання, згідно з теор. 1 та 2.

Розрахуємо основні показники для даного розподілу.

$$S_p = \frac{T_1}{T_{n^3}} = \frac{(2n-1) \cdot n^2}{1 + \log_2 n} \quad (2.6)$$

$$E_p = \frac{S_p}{p} = \frac{\frac{(2n-1) \cdot n^2}{1 + \log_2 n}}{n^3} = \frac{2n-1}{n(1 + \log_2 n)} = \frac{2n-1}{n + \log_2 n^n} \quad (2.7)$$

$$C_p = T_p \cdot p = (1 + \log_2 n) \cdot n^3 \quad (2.8)$$

Далі, визначимо розподіл робіт, при якому кількість процесорів може бути меншою, ніж в попередній задачі. Для цього, візьмемо $p = n$. Використовуючи логіку з попередніх задач, визначимо паралельний час виконання: $T_p = n \cdot (2n-1)$

Розрахуємо основні показники для даного розподілу.

$$S_p = \frac{T_1}{T_n} = \frac{(2n-1) \cdot n^2}{n \cdot (2n-1)} = n \quad (2.9)$$

$$E_p = \frac{S_p}{p} = \frac{n}{n} = 1 \quad (2.10)$$

$$C_p = T_p \cdot p = ((2n-1) \cdot n) \cdot n = (2n-1) \cdot n^2 \quad (2.11)$$

Можемо побачити абсолютно ідентичну ситуацію з усіма показниками (окрім прискорення) з розподілом робіт, де $p = n^2$. Це означає, що цей алгоритм використовує процесори 100% часу роботи алгоритма, а також є вартісно-оптимальним.

2.3 Висновки

Було проведено аналіз розпаралелювання алгоритму добутку двох матриць, розмірністю $n \times n$. Було застосовано декілька конфігурацій з різною кількістю процесорів. Для кожної конфігурації було розраховано основні параметри. Було отримано досвід оцінки трудомісткості паралельних обчислень за допомогою основних теорем.