**Міністерство освіти і науки України**

**Національний університет "Львівська політехніка"**

**Кафедра ЕОМ**

****

**Звіт з розрахункової роботи**

**з дисципліни “Паралельні та розподілені обчислення ”**

**Виконав: студент .гр. КІ-34**

**Алексєєнко Д.С.**

**Прийняв: асистент**

**Козак Н.Б.**

**Львів 2020 р.**

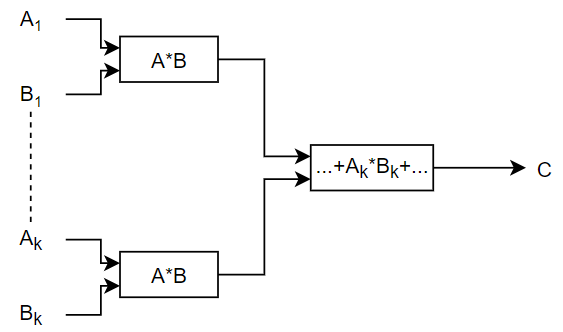
1. 1) Спільна пам’ять передбачає наявність єдиної шини данних для всіх процесорів, тому при роботі з множенням матриць великих розмірів можуть виникати затримки в доступі до пам’яті. Універсальні процесори, тобто ті що готові до будь-яких задач, покажуть нижчі показники швидкодії на відміну від спеціалізованих систем для такого роду обчислень.

2) Спільна пам’ять має ту ж проблему, що й у пункті 1, проте конвеєрна система суматорів, помножувачів та пристроїв ділення дасть значне прискорення при великих об’ємах розрахунків.(найшвидший варіант)

3) Розділена пам’ять внесе затримки так, як для отримання часткових результатів чи інших даних процесорам доведеться обмінюватись повідомленнями з запитами цих диних. Нюанси універсальних процесорів ті ж, що й у пункті 1.(найповільніший варіант)

4) Як і в пункті 3 розділена пам’ять внесе затримки в між-процесорний обмін даними, проте спеціалізована конвеєрна система арифметико-логічних пристроїв покаже свою ефективність при обчисленнях.

***6.***



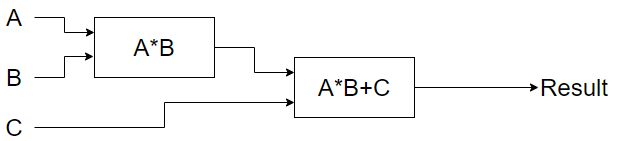
Як видно зі схеми вище – програма ділиться на потоки де знаходяться часткові добутки, після чого вони сумуються у результуючу матрицю/число(залежно від множення стовпець на рядок чи навпаки).

1. Можна.

Для 3 чисел – перше число подати в потік І, друге число подати в потік ІІ,

Паралельно порівняти числа з обох потоків з тим, що залишилось. Якщо знайшлось менше значення – перезаписати число з потоку, інакше порівняти числа в обох потоках і отримати результат.

Для 4 чисел поділити їх попарно на 2 потоки, провести паралельне порівняння в потоках залишивши в кожному потоці менше число. Порівняти залишкові числа між собою і отримати результат.



В даній схемі видно, що в процесі обчислень ми ділимо програму на 2 потоки. В першому потоці отримуємо часткові результати А\*В, а в другому остаточний результат розрахунку А\*В+С.

1. Конвеєр передбачає один робочий потік в якому в певний момент часу відбувається виконання однієї команди і підготовка до виконання іншої. В паралельних системах команди виконуються одночасно.



1. Прискорення S=T1/T100

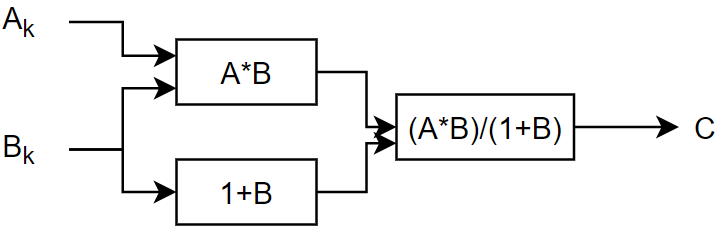
Час паралельного виконання Т100=3T1+((1-3)T1/100), де 3 – відсоток послідовного коду, 100 – кількість процесорів, Т1 – час послідовного виконання програми. => S=T1/2.98T1 => S=1/2.98=0.3356 або 33,56%.

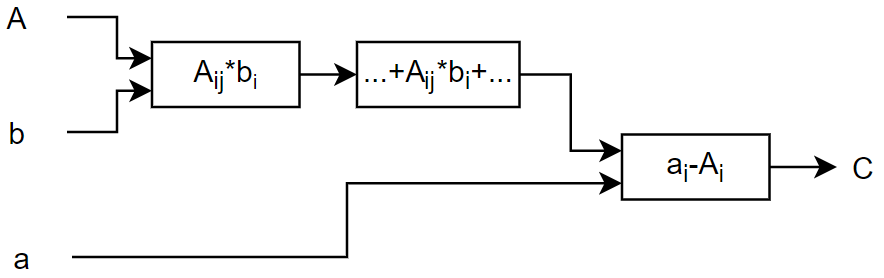
1. Згідно з законом Амдаля програма з 10% послідовних розрахунків не може отримати більше ніж 10-ти кратне прискорення при паралельному режимі роботи.
2. При наявності 20% послідовного коду, а також за умови виконання решти 80% на половині процесорів(10/2=5) => S=0.05 або 5%.
3. Можна. Найбільш ефективний таке накладання буде для гіперкуба розміру d=2 що являтиме собою аналог топології кільце або d=3, в якому прослідковуєтьбся звя’зок схожий на комбінація топологій кільця та повного графа, за вийнятком того, що нумерація вершин в топології гіперкуб різнитиметься на 1 біт.
4. Порядок вузла – величина визначається кількість сусідніх вузлів, де сусідній вузол – той, з яким є пряме з’єднання.

Комунікаційний діаметр мережі – це максимальний шлях між будь-якими двома вузлами.

***7***. У зв’язку з тим, що вираз С=А\*В-В\*С не має значення, так як останній множник С є пустою коміркою результуючої матриці – перетворимо вираз:

A\*B=C+(B\*C) => A\*B=C(1+B) => C = (A\*B)/(1+B)





Як видно зі схеми вище – спершу ми обраховуємо результат множення матриці А на вектор b, де результатом є вектор. Тоді ми проводимо віднімання новоутвореного вектора А від вектора а і отримуємо результат.