Гібридний алгоритм LL^{T}-факторизації розрідженої матриці

Хіміч О.М., Сидорук В.А.

Інститут кібернетики ім. В.М. Глушкова НАН України, пр. Глушкова, 40, Київ, Україна

Анотація. *Розглядається гібридний алгоритм розв'язування систем лінійних алгебраїчних рівнянь з розрідженими симетричними додатно-визначними матрицями на комп'ютерах з графічними прискорювачами. Проведено апробацію алгоритму на гібридному комп'ютері Інпарком-G.*

Abstract. *The hybrid algorithm of untiing of the systems of the linear algebraic evening with sparse symmetric positive-definited matrices on computers with the graphic accelerating is considered. Approbation of algorithm on hybrid computer Inparcom-G is executed.*

1 Вступ

Знаходження розв'язку значної кількості прикладних задач включає як складову частину знаходження розв'язку системи лінійних алгебраїчних рівнянь (СЛАР) з розрідженою матрицею. Характерною особливістю СЛАР, що виникають в тих чи інших задачах є їх великий порядок, а також мала заповненість ненульовими елементами.

Разом з тим вимоги до високопродуктивних обчислень для розв'язання таких задач набагато випереджають можливості традиційних паралельних комп'ютерів, навіть не зважаючи на багатоядерність процесорів. Розв'язання проблеми прискорення обчислень на комп'ютерах з багатоядерними процесорами розглядається в площині використання для прискорення обчислень гібридних систем на основі багатоядерних CPU і GPU.

Пропонується гібридний алгоритм розв'язування СЛАР з розрідженою матрицею на основі методу LL^{T} факторизації.

2 Постановка задачі

Розглянемо задачу

$$Ax = b \qquad (1)$$

з симетричною додатно-визначеною розрідженою матрицею порядку n.

Теоретичною передумовою методу розв'язання задачі (1) для розріджених матриць на комп'ютерах гібридної архітектури є попереднє застосування до вихідної матриці методу паралельних перерізів, який приводить вихідну матрицю до блочно-діагонального вигляду з обрамленням [1, 2]

$$\tilde{A} = P^T A P = \begin{pmatrix} A_{11} & 0 & 0 & \cdots & 0 & A_{1p} \\ 0 & A_{22} & 0 & \cdots & 0 & A_{2p} \\ 0 & 0 & A_{33} & & 0 & A_{3p} \\ \vdots & \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & & A_{p-1p-1} & A_{p-1p} \\ A_{p1} & A_{p2} & A_{p3} & \cdots & A_{pp-1} & A_{pp} \end{pmatrix},$$

де P – матриця перестановок, а блоки A_{ii}, A_{pi} A_{ip} зберігають розріджену структуру, p – кількість діагональних блоків у матриці.

Таким чином, задача розв'язування вихідної задачі (1) зводиться до розв'язування еквівалентної задачі

$$\tilde{A}\tilde{x} = \tilde{b}\,, \qquad (2)$$

де $\tilde{x} = P^{\mathrm{T}} x, \quad \tilde{b} = P^{\mathrm{T}} b$.

Блочний алгоритм LL^{T} розвинення матриці $\tilde{A}$ [3] має наступний вигляд:

$$A_{ii} = L_{ii} L_{ii}^T, \quad L_{ip} = L_{ii}^{-1} A_{ip}, \quad i = \overline{1, p-1},$$

$$A_{pp} = App - \sum_{i=1}^{p-1} L_{pi} L_{ip} = L_{pp} L_{pp}^T, \quad L_{ip} = L_{pi}^T.$$

В результаті матриця $\tilde{L}$ матиме вигляд

$$\tilde{L} = \begin{pmatrix} L_{11} & 0 & 0 & \cdots & 0 & 0 \\ 0 & L_{22} & 0 & \cdots & 0 & \vdots \\ 0 & 0 & L_{33} & & 0 & \vdots \\ \vdots & \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & & L_{p-1} & 0 \\ L_{p1} & L_{p2} & Lp_3 & \cdots & L_{pp-1} & L_{pp} \end{pmatrix},$$

Тоді розв’язання системи (2) зводиться до розв'язання систем

$$\tilde{L} y = \tilde{b}, \qquad \tilde{L}^T \tilde{x} = y. \tag{}$$

3 Гібридний алгоритм

Розглянемо декомпозицію даних для комп’ютерів гібридної архітектури з багатоядерними (CPU) процесорами і графічними (GPU) прискорювачами. Нехай для виконання задачі маємо *p-1* процес. Реалізуємо наступний розподіл даних:

- на GPU, що відповідають процесам з номерами i, $i = \overline{1, p-1}$ зберігаються відповідні діагональні блоки A_{ii}, блоки обрамлення A_{ip} та частини векторів *x, y, b;*
- у GPU, що відповідає процесу з номером 1 додатково зберігається діагональний блок A_{pp}.

Надалі будемо асоціювати процес з графічним прискорювачем.

Враховуючи таку декомпозицію даних запишемо гібридний алгоритм на основі методу LL^T-факторизації.

<u>Факторизація:</u>

У всіх процесах з номерами i, $i = \overline{1, p-1}$ паралельно і незалежно виконуємо наступні операції:

- факторизація блоку: $A_{ii} = L_{ii} L_{ii}^T$, для виконання цього пункту використовується функція, що реалізовує гібридний алгоритм факторизації стрічкової матриці, запропонований у роботі [4].
- модифікація блоку обрамлення $L_{ip} = L_{ii}^{-1} A_{ip}$ де $L_{ip} = L_{pi}^T$, $i = \overline{1, p-1}$.
- знаходження добутку $L_{ip} L_{ip}^T$.
- мультизбирання та модифікація A_{pp} в процесі з номером 1. $\tilde{A}_{pp} = A_{pp} - \sum_{i=0}^{p-1} L_{ip} L_{ip}^T$.
- факторизація модифікованого блоку в першому процесі $\tilde{A}_{pp}$: $\tilde{A}_{pp} = L_{pp} L_{pp}^T$.

<u>Розв’язування системи</u> $\tilde{L} y = \tilde{b}$

У процесах з номерами i, $i = \overline{1, p-1}$ виконуємо наступні операції:

- розв’язування системи $L_{ii} y_i = b_i$;
- знаходження добутків $L_{pi} y_i$.
- мультизбирання та модифікацію p-ї частини вектора *b* в першому процесі $\tilde{b}_p = b_p - L_{pi} y_i$.
- в процесі з номером 1 розв’язування системи $L_{pp} y_p = \tilde{b}_p$.

<u>Розв'язування системи</u> $\tilde{L}^T \tilde{x} = y$

В процесі з номером 1 розв'язуємо систему $L_{pp} x_p = y_p$ і виконуємо розсилку x_p у всі інші процеси.

У процесах з номерами i, $i = \overline{1, p-1}$ виконуємо наступні операції:

- знаходимо добутки $L_{ip} x_p$ і модифікуємо i-у частину вектора y: $y_i = y_i - L_{ip} x_p$.
- розв'язуємо систему $L_{ii} x_i = y_i$.

4 Результати чисельних експериментів

Для реалізації стандартних обчислювальних процедур (множення матриць, розв'язування трикутних систем) використані функції відомих бібліотек, наприклад CUSPARSE, CUSP, Paralution.

Розрахунки проводились на вузлах кластеру Інпарком-G [5], які мають наступні характеристики:

•Процесори: 2 Xeon 5606 (8 ядер) з частотою 2.13 ГГц;
•Графічні прискорювачі: 2 Tesla M2090;
•Об'єм оперативної пам'яті: 24 Гб;
•Комунікаційне середовище: InfiniBand 40 Гбіт/с (з підтримкою GPUDirect), Gigabit Ethernet.
•Також на вузлах встановлена бібліотека MKL 10.2.6 та CUDA починаючи з версії 3.2.

Чисельні експерименти проводились на розріджених матрицях, що вказані в табл. 1. Також в табл. 1 наведені такі характеристики матриці як порядок матриці, кількість ненульових елементів. На рис. 1. показано діаграми, що відображають часи виконання програми на архітектурах 1 CPU, 1 CPU + 1 GPU та n CPU + n GPU. На рис. 2. показано залежність часу виконання програми від кількості використовуваних GPU.

Слід зазначити, що для архітектури n CPU + n GPU параметр **n рівний кількості діагональних блоків мінус один.**

Назва	Порядок матриці	Кількість ненульових елементів
Minsurfo	40806	203622
Dubcova3	146689	3636643
G2_circuit	150102	726674
A22bd_200k	200000	100473400
A22bd_400k	400000	81172000
A22bd_750k	750000	377119000

Табл. 1. Набір тестових матриць.

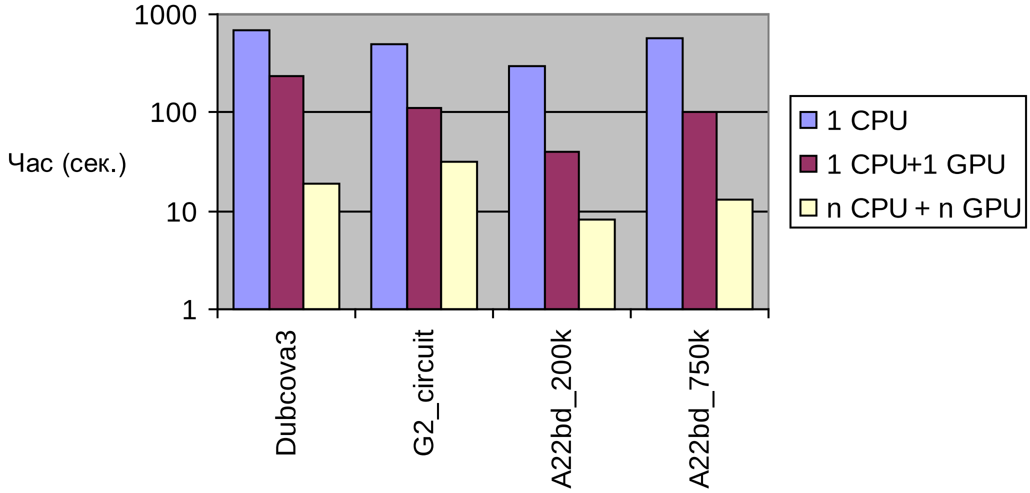

Рис. 1. Часи розв'язання систем з відповідними матрицями

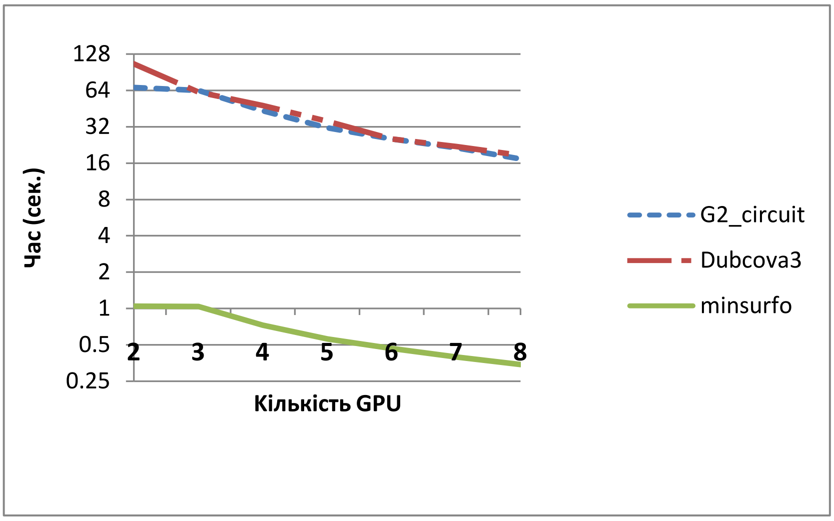

Рис. 2. Залежність часу виконання програми від кількості використовуваних GPU

5 Висновки

Запропоновано гібридний алгоритм на основі методу LL^T факторизації розрідженої симетричної додатно-визначеної матриці. Отримані результати відповідають архітектурам 1 CPU, 1 CPU + 1 GPU, та n CPU + n GPU. Результати експериментів та аналіз програмного забезпечення свідчать, що використання ефективного гібридного алгоритму факторизації СЛАР зі стрічковими матрицями дало змогу скоротити часи реалізації алгоритму на архітектурах 1 CPU + 1 GPU, n CPU+ n GPU; покращенню масштабованості алгоритму і підвищенню ефективності сприятиме реалізація програмного модулю, що дозволить більш ефективно задіяти GPU на етапі розв'язання СЛАР (трикутна стрічкова матриця) з багатьма правими частинами ($L_{ip} = L_{ii}^{-1} A_{ip}$); запропонований алгоритм є найбільш ефективним для стрічкових матриць з широкою стрічкою.

Література

[1] Джордж А., Лю Дж. Численное решение больших разреженных систем уравнений. – М.: Мир, 1984. – 333 с.

[2] Хіміч О.М., Полянко В.В. Оптимізація паралельного ітераційного процесу для лінійних систем з розрідженими матрицями // Теорія оптимальних рішень. – 2011. – № 10 – С. 47 – 53.

[3] Хіміч О.М., Сидорук В.А. Гібридний алгоритм для лінійної задачі найменших квадратів з напіввизначеною розрідженою матрицею // Теорія оптимальних рішень. – 2014. – С. 106-113.

[4] Хіміч О.М., Баранов А.Ю. Гібридний алгоритм розв'язування лінійних систем із стрічковими матрицями прямими методами // Комп'ютерна математика. – 2013. - № 2. С. 80-88

[5] Молчанов И.Н., Химич А.Н., Мова В.И, Николайчук А.А. Интеллектуальный персональный компьютер гибридной архитектуры // Искусственный интеллект. – 2012. – № 3. – С. 73–78