КОМПЬЮТЕРНАЯ
МАТЕМАТИКА

Розглядаються блочні циклічні паралельні алгоритми факторизації дійсних квадратних розріджених невироджених матриць різної структури – стрічкової, профільної, нерегулярної тощо. Запропоновано блочно-циклічний паралельний алгоритм LU-розвинення стрічкової несиметричної матриці.

УДК 519.6

О.В. ПОПОВ

ПРО ПАРАЛЕЛЬНІ АЛГОРИТМИ ФАКТОРИЗАЦІЇ РОЗРІДЖЕНИХ МАТРИЦЬ

Вступ. Математичне моделювання та пов'я-заний з ним комп'ютерний експеримент нині один з основних засобів вивчення різно-манітних явищ і об'єктів природи, процесів і об'єктів у різних галузях науки та техніки, про-цесів у суспільстві, економіці тощо.

При чисельному моделюванні часто вини-кають, наприклад, при використанні триви-мірних моделей, розрахункові (дискретні або напівдискретні) задачі з надвеликою кіль-кістю рівнянь, яка може перевищувати 10^7. Причому дані (матриці) цих задач мають *розріджену* структуру, тобто кількість нену-льових елементів значно менша (не переви-щує 5%) загальної кількості елементів матри-ці. Зберігання таких даних, незважаючи на розрідженість, потребує значних обсягів комп'ютерної пам'яті, які можуть переви-щувати 1 Tb. Задачі такого обсягу вини-кають, зокрема, при використанні тривимір-них моделей.

Зростання параметрів задач, що розв'язуються, розрахунок на комп'ютерах більш повних моделей об'єктів, процесів, явищ вимагає відповідного зростання продуктивності комп'ютерів. Вимоги до високопродуктивних обчислень набагато випереджають можливості персональних комп'ютерів, навіть незважаючи на багатоядерність процесорів.

На сьогодні зростання продуктивності обчислень досягається за рахунок розпаралелювання, яке базується на використанні комп'ютерів з багатьма процесорними пристроями, зокрема з багатоядерними процесорами.

В цих комп'ютерах, як правило, реалізується MIMD-архітектура (архітектура з множинним потоком команд і даних). За останні роки також набули поширення гібридні обчислювальні системи, в яких використовуються сопроцесори, наприклад, графічні процесори, для прискорення обчислень при виконанні великих обсягів однорідних арифметичних операцій. На таких сопроцесорах-прискорювачах, як правило, реалізується SIMD-архітектура паралельних обчислень. Такі комп'ютери гібридної архітектури вже зайняли провідні позиції у світовому рейтингу найпродуктивніших комп'ютерів TOP500 [1].

Розв'язування розрахункових задач у переважній більшості потребує розв'язування систем лінійних алгебраїчних задач (СЛАР) з розрідженими матрицями (див., наприклад, [2, 3]). У свою чергу при розв'язуванні цих СЛАР прямими методами, як правило, використовується факторизація (розвинення) матриці в добуток матриць (в більшості випадків трикутних та діагональних). Часто розвинення матриці в добуток трикутних матриць використовується і в ітераційних методах розв'язування СЛАР.

Структура розріджених матриць визначається нумерацією невідомих і може бути регулярною (наприклад, стрічковою) або нерегулярною.

Мета цієї роботи – дослідження алгоритмів паралельних і розподілених обчислень для факторизації розріджених матриць різної структури. Зокрема, узгодження структури розрідженої матриці та архітектури обчислювальної системи з метою ефективної реалізації алгоритму.

Постановка задачі. Розглянемо СЛАР

$$Ax = b, \tag{1}$$

де в загальному випадку A – дійсна квадратна матриця порядку n; b – матриця правої частини розміру $n{\times}q$ (або n-вимірний вектор). Розв'язування цієї СЛАР полягає у знаходженні такого розв'язку x (матриці розмірності $n{\times}q$ або n-вимірного вектора), щоб рівняння (1) перетворювалося в тотожність. Якщо матриця A системи невироджена (тобто її визначник $|A| \equiv \det(A) \neq 0$), то розв'язок СЛАР (1) існує і єдиний. Якщо матриця симетрична, то розглядатимемо випадок, коли вона додатно визначена.

Розглядаються СЛАР як з симетричними, так і з несиметричними розрідженими матрицями. Тобто, як зазначалося вище, з матрицями, загальна кількість *ненульових* елементів (H_A) яких значно менша загальної кількості всіх елементів (що дорівнює n^2).

Структури розріджених матриць. Структура розрідженої матриці визначається нумерацією невідомих і може бути регулярною (стрічковою, профільною, блочно-діагональною з обрамленням тощо) або нерегулярною (див. рис. 1). Зміна нумерації невідомих викликає зміну структури матриць, яку можна таким чином оптимізувати за якимось критерієм.

Позначимо $\eta_C(i)$ – кількість ненульових елементів в i-у рядку матриці C, $\mathrm{H}_A = \sum_{i=1}^{n} \eta_A(i)$, $m_l(i) = i - \min\{j \mid a_{ij}{\neq}0\}$, $m_u(i) = \max\{j \mid a_{ij}{\neq}0\} - i$, $m_l = \max_{1 \le i \le n} m_l(i)$, $m_u = \max_{1 \le i \le n} m_u(i)$. Зауважимо, що у випадку симетричної матриці $m_l = m_u = m$. Множини значень $m_l(i)$ та $m_u(i)$, $i = 1, 2, \ldots, n$, визначають профіль матриці A.

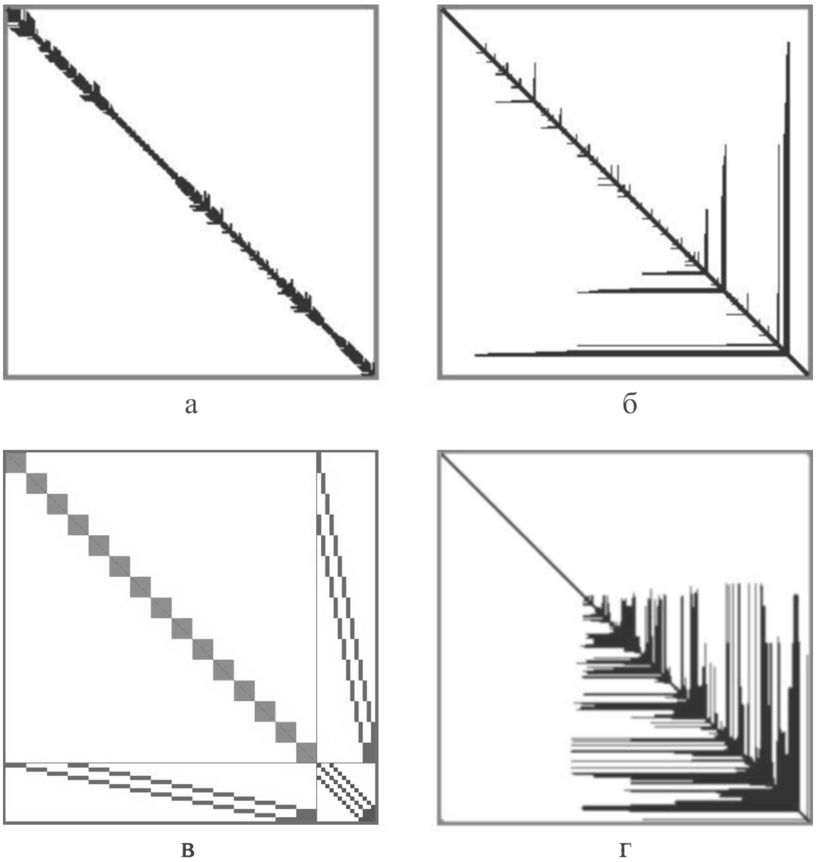

РИС. 1. Приклади структур розріджених матриць

Серед різних розріджених матриць доцільно виділити матриці, ненульові елементи яких концентруються біля головної діагоналі (рис. 1, *а*, 1, *б*), тобто коли $m_l \ll n$ та $m_u \ll n$. Такі матриці часто називають профільними. А якщо $\min\limits_{m_l < i \le n} m_l(i)$ та $\min\limits_{1 \le i \le n - m_u} m_u(i)$ мало відрізняються (на 5–10 %) відповідно від m_l і m_u, то таку матрицю доцільно вважати стрічковою з шириною стрічки $m_l + m_u + 1$. Якщо ширина стрічки не перевищує 300, то матрицю вважають вузькою стрічковою. Розріджені матриці з таким структурами переважно є вихідними або отримуються в результаті оптимізації за алгоритмами фактор-дерев або Катхілл – Маккі [2].

При використанні алгоритму паралельних перерізів [2] для оптимізації структури матриці отримують матрицю із структурою, яку показано на рис. 1, *б*, 1, *в* і яку часто називають блочно-діагональною з обрамленням. У цьому випадку ненульові елементи розташовуються в досить великих діагональних блоках та в блоках обрамлення. Часто також доцільно оптимізувати структуру цих діагональних блоків.

Алгоритм мінімальної степені [2] дозволяє суттєво зменшити загальну кількість елементів у профілі матриці, але оптимізована структура матриці втрачає регулярність і в ній з'являються довгі рядки (або стовпчики) такі, що m_l та / або m_u можуть перевищувати $0{,}5n$ (рис. 1, *г*).

Розв'язування СЛАР з розрідженими матрицями. Перші детальні дослідження методів розв'язування СЛАР з розрідженими матрицями почали проводитися з появою і розвитком ЕОМ. Через обмежені можливості ЕОМ 50–70 років минулого століття спочатку розвивалися ітераційні методи. Проте як збіжність, так і ефективність ітераційних методів істотно залежить від знання апріорної інформації про властивості матриці задачі. Крім того в разі багаторазового розв'язування СЛАР з однією і тією ж матрицею і різними правими частинами прямі методи виявляються набагато ефективнішими. Тому в міру нарощування обчислювальних ресурсів комп'ютерів все більш широкий розвиток отримують алгоритми прямих методів розв'язування СЛАР з розрідженими матрицями.

У 80-і роки минулого століття опубліковано низку фундаментальних робіт, які висвітлювали основні досягнення в цьому напрямку. В монографії [2] описана більшість основних методів розв'язування СЛАР з розрідженими симетричними додатно визначеними матрицями великих розмірів, а монографія [3] присвячена проблемам реалізації наведених методів на паралельних комп'ютерах. На основі накопиченого матеріалу визначено низку ефективних алгоритмів розв'язування СЛАР з розрідженими матрицями, способи зберігання розріджених даних, а також різні алгоритми оптимізації заповнення (див., наприклад, [4], де наведено широкий огляд сучасних алгоритмів розв'язування СЛАР з розрідженими матрицями).

При використанні прямих методів матриця задачі факторизується в добуток трикутних або трикутних та діагональної матриць. Також варто зазначити, що збіжність ітераційних методів залежить від використовуваної схеми (явної або неявної) і є невисокою для явних схем. Використання неявних схем з передобумовлювачем дозволяє зменшити кількість ітерацій, але потребує, як правило, факторизації матриці-передобумовлювача, яка також має розріджену структуру.

Отже, розв'язування системи (1) полягає у розв'язуванні трьох підзадач:

– розвинення (факторизація) матриці системи

$$A = LU, \quad \text{або} \quad A = LL^{\mathrm{T}}, \quad \text{або} \quad A = LDL^{\mathrm{T}}, \tag{2}$$

де L – нижня трикутна матриця з одиницями на головній діагоналі (крім випадку LL^T-розвинення), U – верхня трикутна матриця, D – діагональна матриця;

– розв'язування СЛАР з нижньою трикутною матрицею

$$Ly = b;$$

– розв'язування СЛАР з верхньою трикутною матрицею

$$Ux = y, \quad \text{або} \quad L^{\mathrm{T}}x = y, \quad \text{або} \quad L^{\mathrm{T}}x = D^{-1}y.$$

Розвинення розріджених матриць. Розвинення симетричних матриць проводиться методом Холецького, а розвинення несиметричних матриць – методом Гаусса. При цьому у випадку несиметричних матриць, якщо відсутня діагональна перевага (коли модуль кожного діагонального елемента перевищує суму модулів позадіагональних елементів відповідного рядка або стовпчика), необхідно проводити хоча б частковий вибір головного елемента.

Наприклад, елементи LDL^T-розвинення з (2) можна обчислити за формулами (для $k = 1, 2, \ldots, n$):

$$l_{kj} = t_{kj}/d_j,\ \ j=1,\ldots,k-1, \quad d_k = a_{kk} - \sum_{j=1}^{k-1} l_{kj} t_{kj}\,, \quad t_{ik} = a_{ik} - \sum_{j=1}^{i-1} l_{kj} t_{ij}\,, \quad i = k+1,\ldots,n. \tag{3}$$

LU-розвинення несиметричної матриці може бути виконане за формулами (для $i, j = k+1, \ldots, n$, $k = 1, 2, \ldots, n$):

$$u_{kk} = a_{kk}^{(k-1)}, \quad u_{kj} = a_{kj}^{(k-1)}, \quad m_{ik} = a_{ik}^{(k-1)} / u_{kk}, \quad a_{ij}^{(k)} = a_{ij}^{(k-1)} - u_{kj} m_{ik}\,. \tag{4}$$

Тут m_{ij} – елементи матриці L, причому у випадку виконання розвинення без вибору головного елемента $m_{ij} \equiv l_{ij}$. З формули (4) легко отримати формули для LDL^T-розвинення симетричної матриці алгоритму метода Холецького у формі зовнішніх добутків (для $i = k+1, \ldots, n$, $k = 1, 2, \ldots, n$):

$$d_k = a_{kk}^{(k-1)}, \quad l_{ik} = a_{ik}^{(k-1)} / d_k, \quad a_{ij}^{(k)} = a_{ij}^{(k-1)} - l_{ik} a_{jk}^{(k-1)}, \quad j = k+1, \ldots, i. \tag{5}$$

У випадку розріджених матриць загальний обсяг обчислень істотно скорочується, якщо обчислення за формулами або (3), або (4), або (5) виконувати тільки з явно ненульовими елементами розвинення матриці.

Розглянемо, наприклад, обчислення за формулами (3). Позначимо $Z_i(L)$ – множина других індексів ненульових піддіагональних елементів i-го рядка нижньої трикутної матриці L. Тоді підсумовування в формулах (3) ведеться лише для $k \in Z_i(L)$ або $k \in Z_i(L) \cap Z_j(L)$. Таким чином, елемент $l_{ij} = 0$, якщо

$$a_{ij} = 0 \ \text{ та } \ Z_i(L) \cap Z_j(L) = \varnothing. \tag{6}$$

Якщо в рядках нижньої трикутної матриці L міститься багато довільно розташованих нульових елементів, то при обчисленнях за формулами (3) суттєво зростають накладні витрати на пошук відповідних пар ненульових елементів. У цьому випадку доцільно використовувати інший порядок обчислень елементів трикутного розвинення – за формулами (5).

У випадку несиметричної матриці маємо аналогічні (6) умови того, що $a_{ij}^{(k)} = 0$, а отже і $l_{ij} = 0$ або $u_{ij} = 0$,

$$a_{ij} = 0 \ \text{ та } \ Z_i(L) \cap Z_j(U) = \varnothing. \tag{7}$$

Легко бачити, що всі нульові елементи вихідної матриці, які розташовано в рядках ліворуч або в стовпчиках вище за перший ненульовий елемент залишаються нульовими у відповідному трикутному розвиненні. Тому ці елементи в обчисленнях участь не беруть, а перші ненульові елементи – ліві (в рядках) або верхні (в стовпчиках) – визначають профіль розрідженої матриці. Формули (6), (7) свідчать, що профілі вихідної матриці та її трикутного розвинення збігаються. Водночас заповнення профілю трикутного розвинення розрідженої матриці суттєво зростає у порівнянні з вихідною розрідженою матрицею. Заповнення профілю трикутного розвинення розрідженої матриці залежить від зв'язків між невідомими системи і регулюється їх нумерацією. Існують (див., наприклад [2]) різні алгоритми переупорядковування невідомих, які дозволяють як зменшити заповнення, так і поліпшити інші характеристики профілю трикутного розвинення, які впливають на загальну кількість арифметичних операцій при розв'язуванні СЛАР.

Паралельні алгоритми трикутного розвинення розріджених симетричних матриць. Паралельні алгоритми дослідження та розв'язування задач лінійної алгебри з симетричними матрицями, в тому числі трикутного розвинення розрідженої симетричної матриці, досить повно представлено в монографії [5].

Історично перші паралельні алгоритми трикутного розвинення розрідженої симетричної матриці розроблено для блочно-діагонального з обрамленням представлення такої матриці [6]. Досить детально такий же алгоритм описано також в [5] для розв'язування задач з вузькими стрічковими матрицями. Проте можливості паралельних алгоритмів цієї групи обмежені через появу в процесі розвинення зведеної матриці, порядок якої дорівнює кількості рядків (стовпчиків) в обрамленні. Трикутне розвинення такої матриці може бути виконано або в послідовному режимі (якщо її порядок порівняно невеликий), або в паралельному, але для цього необхідно перерозподілити дані між процесорними пристроями та використати паралельний алгоритм для щільних симетричних матриць. Крім того, при розвиненні матриць з вузькою стрічкою необхідно таку матрицю, використовуючи алгоритм паралельних перерізів, привести до блочно-діагональної матриці з обрамленням, що збільшує в 4 рази кількість арифметичних операцій. Тому ефективність такого алгоритму обмежена зверху 25 %.

У роботі [7] запропоновано рядково-циклічний паралельний алгоритм методу Холецького для стрічкових симетричних матриць, в якому реалізовано формули (3). Надалі на базі цього алгоритму розроблено одновимірний блочний циклічний алгоритм методу Холецького для стрічкових і профільних симетричних матриць (див., наприклад [5]). Цей алгоритм дозволяє проводити обчислення лише з явно ненульовими елементами матриць (вихідної та трикутного розвинення. Проте накладні витрати на пошук таких елементів можуть перевищити виграш від скорочення кількості арифметичних операцій.

Ефективне використання одновимірних блочних циклічних алгоритмів можливо при високій збалансованості завантаження процесів, щоб не було простоїв окремих процесів при виконанні обчислень. У загальному випадку цього можна досягти для стрічкових матриць. Якщо ж кількість елементів, з якими проводяться операції, в рядках профілю матриці істотно відрізняються, то обчислення можуть потребувати часу, який порівнянний з випадком стрічкової матриці з шириною стрічки, що дорівнює максимальній кількості елементів профілю в одному рядку.

Для розріджених симетричних матриць довільної структури часто можна досягти кращої збалансованості, якщо виконувати розвинення за формулами (5). У цьому випадку при використанні схеми завдання ненульових елементів групами і одновимірного блочного циклічного або рядково-циклічного розподілу даних (верхнього трикутника вихідної матриці та верхньої трикутної матриці розвинення) між процесами можна досягти більш рівномірного завантаження процесів та подолати «проблему довгих рядків», яка згадувалась вище. Відповідний одновимірний блочний циклічний алгоритм запропоновано в [8].

В роботах [5, 8] досліджено ефективність запропонованих блочного та одновимірних блочних циклічних паралельних алгоритмів факторизації розріджених симе-тричних матриць. У роботі [8] доведено, що ефективність цих паралельних алгоритмів обмежена зверху $(O_1 = \sum_{i=1}^{n} \eta_L^2(i))$.

$$E_p \le 1 - \left(\frac{p-1}{O_1} \sum_{i=1}^{n} \eta_L(i) + \frac{p \log_2 p}{O_1} \left(\frac{t_c}{t} \frac{n}{s} + \frac{t_o}{t} \sum_{i=1}^{n} \eta_L(i) \right) \right), \tag{8}$$

де p – кількість паралельних процесів, s – розмір блоку, t – середній час виконання однієї арифметичної операції (додавання або множення) з плаваючою комою, t_o – час, необхідний для обміну одним машинним словом між двома процесами, t_c – час, необхідний для синхронізації двох процесів. Виходячи з оцінки (8), для стрічкових матриць маємо оцінку [5]:

$$E_p \approx 1 - \left(\frac{2(p-1)(s+1)}{m} + \frac{p \log_2 p}{m} \left(\frac{t_o}{t} + \frac{1}{sm} \frac{t_c}{t} \right) \right),$$

а для блочного алгоритму розвинення вузької стрічкової матриці маємо [5]:

$$E_\text{p} \approx 0{,}25 - \frac{1}{8m} - \frac{p-1}{16n} \left(\frac{7p-18}{3} m + \frac{p}{2} \left(\frac{t_\text{o}}{t} + \frac{2}{m^2} \frac{t_\text{c}}{t} \right) \right).$$

Враховуючи сучасні тенденції розвитку технічних та програмних засобів для високопродуктивних обчислень, зокрема розробку виробниками технічних засобів також і програмного забезпечення, доцільно модифікувати запропоновані алгоритми так, щоб переважну більшість операцій зводилась до матрично-матричних або матрично-векторних операцій із щільними блоками елементів розріджених матриць. Такий підхід буде використано далі для випадку несиметричної матриці.

Блочний циклічний паралельний алгоритм розвинення стрічкової несиметричної матриці. Формули (5) LU-розвинення стрічкової несиметричної матриці з частковим вибором по стовпчику головного елемента можна записати в блочній формі. Для $K = 1, 2, \ldots, N$:

$$A_K^{(K-0,5)} = P^{(K)} A_K^{(K-1)},$$

$$L_{I,K}^{(K)} U_{K,K} = A_{I,K}^{(K-0,5)}, \quad I = K, K+1, \ldots, \min\{K + \mathrm{M}_l, N\},$$

$$L_{K,K}^{(K)} U_{K,J} = A_{K,J}^{(K-0,5)}, \quad J = K+1, \dots, \min\{K + \mathrm{M}_u, N\},$$
$$A_{I,J}^{(K)} = A_{I,J}^{(K-0,5)} - L_{I,K}^{(K)} U_{K,J}, \quad I = K+1, \dots, \min\{K + \mathrm{M}_l, N\},$$
$$J = K+1, \dots, \min\{K + \mathrm{M}_u, N\}, \tag{9}$$

де $N = \lceil (n-1)/s \rceil + 1$ (s – розмір блока, $\lceil a \rceil$ – ціла частина числа a), $M_l = \lceil (m_l - 1)/s \rceil + 1$, $M_u = \lceil (m_u + m_l - 1)/s \rceil + 1$, $A_K^{(K-1)}, A_K^{(K-0,5)}$ – права нижня підматриця порядку $n - (K-1)s$, $A_0^{(0)} \equiv A$, $P^{(K)}$ – матриця перестановок на K-у кроці, $A_{I,J}^{(K)}$, $L_{I,J}^{(J)}, U_{I,J}$ – у загальному випадку (крім, можливо, останніх рядка та стовпчика блоків) квадратні блоки порядку s. За рахунок перестановок збільшується порівняно з матрицею A кількість наддіагоналей верхньої трикутної матриці U – до $m_u + m_l$. Водночас для економії обчислень на K-у кроці не виконуються перестановки в матрицях $L_{I,J}^{(J)}$ при $I \geq K$ та $J < K$. Необхідно зауважити, що згідно (9) на K-у кроці модифікується тільки прямокутна підматриця розмірами $(m_l+s) \times (m_u+m_l+s)$ (рис. 2).

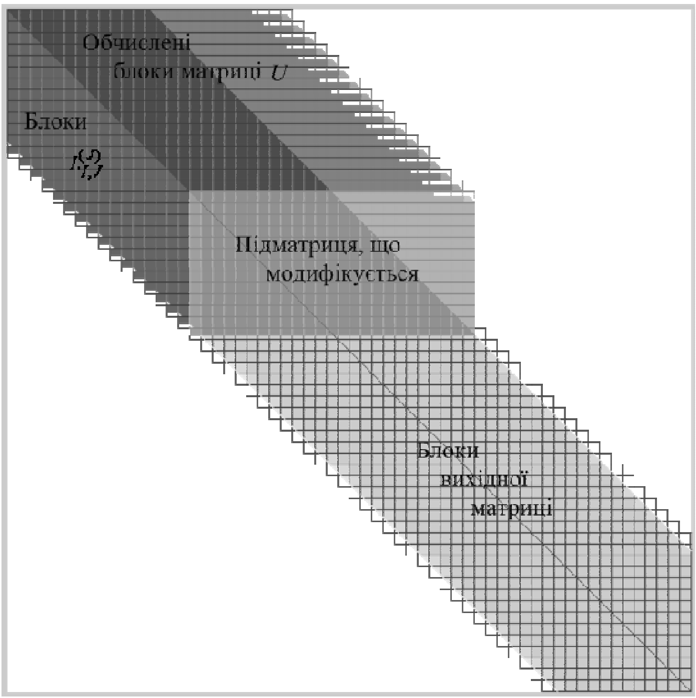

РИС. 2. Схема блочного алгоритму

Аналіз формул (9) засвідчив, що для реалізації більшості обчислень (крім обчислення блоків $L_{I,K}^{(K)}$) можна використати програмні модулі для матрично-матричних операцій від розробників технічних засобів. Ці модулі можуть ефективно виконуватись також на прискорювачах-сопроцесорах.

Для паралельного алгоритму рядки блоків вихідної матриці A та матриць розвинення L, U розподіляються циклічно між процесами так, щоб кожен процес мав хоча б один рядок блоків, які модифікуються на даному кроці.

Називатимемо на K-у кроці провідним рядком або стовпчиком блоків відповідно K-й рядок або стовпчик, а провідним процесом процес, якому розподілено провідний рядок блоків. Тоді блочно-циклічний паралельний алгоритм полягає у наступній послідовності операцій. В циклі по $K = 1, 2, \ldots, N$.

1. Формування підматриці, що модифікується $A_K^{(K-0,5)}$ та обчислення блоків розвинення $U_{K,K}^{(K)}$ і $L_{I,K}^{(K)}$, $I = K, \ldots, K+M_l$.

2. Обчислення згідно (9) блоків розвинення $U_{K,J}$ провідного рядка; виконується провідним процесом.

3. Розсилка всім процесам провідного рядка блоків.

4. Модифікація (s-рангова) згідно (9) елементів блоків $A_{I,J}^{(K)}$ прямокутної підматриці, що модифікується; виконується паралельно всіма процесами.

Порядок і ефективність виконання першого кроку переважно залежить від множини елементів стовпчика матриці, на якій виконується вибір головного елемента. Найбільш ефективним є варіант, коли головний елемент вибирається тільки серед елементів стовпчика, які розподілено провідному процесу. Тоді провідний процес формує свою частину підматриці $A_K^{(K-0,5)}$, обчислює блоки розвинення $L_{K,K}^{(K)}$ і $U_{K,K}^{(K)}$ та розсилає блок $U_{K,K}^{(K)}$ іншим процесам, після цього всі процеси обчислюють згідно (9) розподілені їм блоки розвинення $L_{I,K}^{(K)}$, $I = K, \ldots, K+M_l$.

Висновок. Запропоновані блочно-циклічні паралельні алгоритми факторизації розріджених матриць різної структури є досить ефективними, що підтвердили результати їх апробації на різних кластерних комплексах [5, 7, 8]. Подальший їх розвиток пов’язаний з використанням програмних модулів від виробників технічних засобів, що реалізують матрично-векторні та матрично-матричні операції, у тому числі на комп’ютерах гібридної архітектури. Актуальним є розповсюдження такого підходу із стрічкових матриць на матриці інших розріджених структур.

Роботу виконано за договором В.К.150.12.13 «Розробка та впровадження інтелектуальних інформаційних технологій грід-обчислень для математичного моделювання процесів, що протікають при зварюванні та споріднених процесах» в рамках Державної цільової науково-технічної програми «Впровадження і застосування грід-технологій на 2009-2013 роки».

А.В. Попов

О ПАРАЛЛЕЛЬНЫХ АЛГОРИТМАХ ФАКТОРИЗАЦИИ РАЗРЕЖЕННЫХ МАТРИЦ

Рассматриваются блочные циклические параллельные алгоритмы факторизации невырожденных действительных квадратных разреженных матриц различной структуры – ленточной, профильной, нерегулярной и др. Предложен блочно-циклический параллельный алгоритм *LU*-разложения ленточной несимметричной матрицы.

A.V. Popov

ON THE PARALLEL ALGORITHMS OF SPARSE MATRICES FACTORIZATION

The block cyclic parallel algorithms of real square sparse nonsingular matrices of different structure (band, profile, irregular and others like that) decomposition are dealt with. The block cyclic parallel algorithm of *LU*-decomposition of band asymmetrical matrix is offered.

1. http://www.top500.org
2. *Джордж А., Лю Дж.* Численное решение больших разреженных систем уравнений. – М.: Мир, 1984. – 334 с.
3. *Ортега Дж.* Введение в параллельные и векторные методы решения линейных систем – М.: Мир, 1991. – 300 с.
4. *Глушакова Т.Н., Эксаревская М.Е.* Методы работы с разреженными матрицами произвольного типа. – Воронеж: Воронежский гос. ун-т, 2005. – 44 с.
5. *Химич А.Н., Молчанов И.Н., Попов А.В. и др.* Параллельные алгоритмы решения задач вычислительной математики – Киев: Наукова думка, 2008. – 248 с.
6. http://www.netlib.org/scalapack
7. *Попов А.В., Химич А.Н.* Параллельный алгоритм решения системы линейных алгебраических уравнений с ленточной симметричной матрицей // Компьютерная математика. – 2005. – № 2. – С. 52–59.
8. *Химич А.Н., Попов А.В., Полянко В.В.* Алгоритмы параллельных вычислений для задач линейной алгебры с матрицами нерегулярной структуры // Кибернетика и системный анализ. – 2011. – № 6. – С. 159–174.

Одержано 11.10.2013

Про автора:

Попов Олександр Володимирович,
кандидат фізико-математичних наук, старший науковий співробітник

Інституту кібернетики імені В.М. Глушкова НАН України.