УДК 004.42

Лазорко Н. В., магістр, група ПІ-49м,
Яремчук С. І., канд. фіз.-мат. наук, проф.
Житомирський державний технологічний університет

ПАРАЛЕЛЬНІ ОБЧИСЛЕННЯ В АЛГОРИТМІ ГОМОРІ РОЗВ'ЯЗАННЯ ЗАДАЧІ РОЗМІЩЕННЯ ДЖЕРЕЛ ФІЗИЧНОГО ПОЛЯ

Задача розміщення джерел фізичного поля на фіксовані місця можна описати наступним чином. Є область $\Omega \subset R^n$; N джерел фізичного поля $D_i, i \in [1:N]$; N посадкових місць $n^j \in \Omega, j \in [1:\mathrm{N}]$ та K контрольних точок. Необхідно розмістити джерела фізичного поля на посадкові місця таким чином, щоб максимальне із значень поля в контрольних точках було найменшим. Кожне джерело повинно займати одне посадкове місце та на одне посадкове місце повинно призначатися лише одне джерело. Відповідно до змістовної постановки задачі була побудована математична модель (1-3). Керовані змінні.

$$x_{ij} = \begin{cases} 0, \textit{якщо}\, i - \textit{те джерело не призначається на}\ j - \textit{те місце} \\ 1, \textit{якщо}\, i - \textit{те джерело призначається на}\ j - \textit{те місце} \end{cases}$$

Обмеження:

$$\begin{cases} \sum_{i=1}^{N} x_{ij} = 1, j \in [1:\mathrm{N}], \\ \sum_{j=1}^{N} x_{ij} = 1, i \in [1:\mathrm{N}], \end{cases} \tag{1}$$

$$x_{ij} \in \{0,1\}, i \in [1:\mathrm{N}], j \in [1:\mathrm{N}], \tag{2}$$

Функція цілі:

$$f(x) = \max_{k \in [1:K]} f_k(x) \to \min, \tag{3}$$

де $f_k(x) = \sum_{i=1}^{N} \sum_{j=1}^{N} c_{ij}^k x_{ij}$, c_{ij}^k – вклад джерела, що знаходиться на посадковому місці, в значення поля в контрольній точці.

На практиці часто розв'язуються задачі великої розмірності. Тобто, якщо є n – обмежень та n-посадкових точок, то розмірність такої задачі вже буде досягати n2. Для підвищення ефективності обрахунків задач, використовуються різні підходи та способи розпаралелення, тобто відбувається розбиття алгоритму на блоки або окремі гілки, що передаються процесорам та можуть обраховуватися незалежно один від одного. Щоб вдосконалити алгоритм Гоморі для розв'язання задач розміщення джерел фізичного поля на фіксовані місця, було проаналізовано та досліджено способи розбиття процесорної здатності комп'ютера та обрано той, що може бути застосований до даного алгоритму. Підхід, що був обраний, має назву алгоритм Фокса, що включає в себе конвеєр та паралельні операції. Його основні принципи – це: розпаралелення суттєво послідовних операцій. В даному випадку, це перехід від однієї симплекс таблиці до іншої, тобто пошук кутової точки на виході. За таким же принципом відбувається і побудова відсічення на кожному кроці; з'єднання процесорів таким чином, щоб результат роботи одного процесора потрапляв на вхід іншого (лінійна топологія); розбиття складної операції на декілька послідовних стадій, кожна з якої виконується своїм процесором.

Маємо основний алгоритм: процесор 1 передає дані процесору 2; процесор 2 додає свої дані і передає процесору 3 і так далі до отримання розв'язку ЗЛП. Результат передається на початок процесору 1; процесор 1 починає перевірку умови цілочисельності. Якщо розв'язок не задовольняє умовам, то застосовується паралельна реалізація, де кожен процесор обраховує свою дію, а саме шукає дробову частину вільного члена того рядка, що не відповідає умовам цілочисельності. І по закінченню кожен процесор віддає свої результати обрахунків, з яких обирається максимальна дробова частина; всі результати розрахунків кожного з процесорів повертаються назад до процесора 1, а саме в результаті отримуються розв'язуючий рядок та розв'язуючий елемент; будується правильне відсічення та процес знову повторюється до отримання оптимального розв'язку задачі: цілочисельного або частково цілочисельного.

Аналіз ефективності алгоритму. Час паралельних розрахунків складається з часу роботи процесора та часу виконання передачі (1):

$$k_e = \frac{\mathrm{T}_1}{T_p} = \frac{1}{\frac{q}{n} + \frac{2q(p-1)\tau_0 + \tau_c \log_2 p}{2n^2 - n}} \qquad (1)$$

Час вирішення m задач на одному процесорі для конвеєра буде мати вигляд: $T_1 = mpt$. А його прискорення (2):

$$k_e = \frac{\mathrm{mpt}}{(p-1)t_0 + pt + (m-1)(t+t_0)} \qquad (2)$$

Дослідивши алгоритми розпаралелення, можна зробити висновки, що алгоритм Фокса, порівняно з іншими: достатньо ефективний при великих розмірностях матриці; можливість розпаралелення принципово послідовних операцій; можливість одночасного виконання передачі даних та їх обробки (асинхронні операції); в момент використання паралелізації потребує синхронізації, що знижує ефективність. Але в даному випадку обрахунки не містять великого об'єму даних.