computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЯ ПАРАЛЕЛЬНОГО ДОСТУПУ ДО ІНФОРМАЦІЇ ФАЙЛІВ БАЗ ДАНИХ ДЛЯ БАГАТОПРОЦЕСОРНИХ ЕОМ

Володимир Лісовець, Григорій Цегелик

Львівський національний університет імені Івана Франка,
вул. Університетська, 1, Львів, 79000, e-mail: kafmmsep@franko.lviv.ua

Резюме: *в статті розглянуто метод m-паралельного послідовного перегляду та два варіанти методу m-паралельного блочного пошуку, орієнтовані на їх використання в багатопроцесорних системах для пошуку інформації у файлах баз даних. Досліджено ефективність цих методів для відомих законів розподілу ймовірностей звертання до записів. За критерій ефективності взято математичне сподівання кількості паралельних порівнянь, необхідних для пошуку запису у файлі. Проведено порівняльний аналіз ефективності методів і для кожного розглянутого закону розподілу ймовірностей звертання до записів визначено свій найкращий метод. Побудовано оптимальні стратегії пошуку записів в послідовних файлах, які зберігаються у зовнішній пам'яті багатопроцесорної ЕОМ. За критерій оптимальності прийнято математичне сподівання загального часу, необхідного для пошуку запису у файлі.*

Ключові слова: *багатопроцесорні системи, математичне моделювання, паралельний пошук, бази даних.*

ВСТУП

Створення паралельних обчислювальних систем є стратегічним напрямком розвитку обчислювальної техніки. Це викликано обмеженістю максимально можливої швидкодії звичайних послідовних ЕОМ, а також наявністю обчислювальних задач, для розв'язування яких можливості існуючих засобів обчислювальної техніки недостатні.

Проблема створення високопродуктивних обчислювальних систем належить до переліку найскладніших науково-технічних задач. Організація паралельних обчислень здійснюється, в основному, за рахунок уведення надлишкових функціональних пристроїв (декількох процесорів). Якщо здійснити поділ алгоритмів, які застосовуються, на інформаційно-незалежні частини й організувати виконання кожної частини обчислень на різних процесорах, то можна прискорити процес обчислень. Такий підхід дозволяє виконувати необхідні обчислення з меншими затратами часу. Одержання максимально-можливого прискорення обмежується тільки кількістю наявних процесорів і "незалежних" частин алгоритму.

Завдяки високій надійності та продуктивності багатопроцесорні ЕОМ широко використовуються для підтримки й організації великих баз даних (БД). При розв'язуванні різноманітних задач із використанням БД основний акцент переноситься з процедур обробки інформації на процедури організації збереження та пошуку інформації в них. Тому продуктивність обчислювальних систем, орієнтованих на роботу з великими БД, у значній мірі визначається ефективністю методів паралельного пошуку інформації в БД.

1. МЕТОДИ ПАРАЛЕЛЬНОГО ПОШУКУ ТА ЇХ ЕФЕКТИВНІСТЬ

Розглядаються наступні методи паралельного пошуку записів у файлах БД для багатопроцесорних ЕОМ [1-5]:

- метод m-паралельного послідовного перегляду;
- перший варіант методу m-паралельного блочного пошуку;
- другий варіант методу m-паралельного блочного пошуку.

Проводиться аналіз ефективності цих методів для різних законів розподілу ймовірностей звертання до записів (рівномірного, "бінарного", Зіпфа й узагальненого, частковим випадком якого є розподіл, що наближено задовольняє правило "80-20" [6-9]). За критерій ефективності приймається математичне сподівання кількості паралельних порівнянь, необхідних для пошуку запису у файлі.

Зауважимо, що у випадку однопроцесорних ЕОМ ефективність методів пошуку для різних законів розподілу ймовірностей звертання до записів, а також порівняння методів за ефективністю досліджено в [10]. Деякі часткові випадки ефективності методів розглянуто в [6, 7].

Дослідження ефективності методів паралельного пошуку проведено для рівномірного розподілу ймовірностей звертання до записів і таких законів нерівномірного розподілу ймовірностей, як:

- "бінарний" розподіл

$$p_i = \frac{1}{2^i},\ i = \overline{1, N-1},\ p_N = \frac{1}{2^{N-1}},$$

де pi – ймовірність звертання до i-го запису, N – кількість записів у файлі;

- закон Зіпфа

$$p_i = \frac{1}{iH_N},\ i = \overline{1, N},$$

де $H_N = \sum_{k=1}^{N} \frac{1}{k}$;

- узагальнений закон розподілу

$$p_i = \frac{1}{i^{(c)} H_N^{(c)}},\ i = \overline{1, N},$$

де c $(0 < c < 1)$ – будь-який параметр і $H_N^{(c)} = \sum_{k=1}^{N} \frac{1}{k^c}$.

Розглянемо метод m-паралельного послідовного перегляду.

Припустимо, що до складу багатопроцесорної ЕОМ входять m процесорів, які працюють паралельно та мають спільне поле пам'яті. Пронумеруємо процесори натуральними числами від 1 до m. Суть методу m-паралельного послідовного перегляду полягає в такому. Розіб'ємо всі записи файлу умовно на блоки по m записів у кожному. Нехай $N = n \cdot m$ – кількість записів у файлі, де n – кількість блоків. Тоді при використанні m-паралельного послідовного перегляду процес пошуку запису буде складатися з низки кроків. На першому кроці i-ий процесор переглядає значення ключа i-го запису. При цьому процес перегляду може бути успішним або неуспішним. Для визначення "успішності" всі процесори повинні обмінятися інформацією. У разі успішного перегляду процес пошуку завершується. Якщо перегляд неуспішний, то на другому кроці i-ий процесор переглядає значення ключа $(m + i)$-го запису і т. д. На $(k + 1)$-му кроці (у випадку неуспішного перегляду на k-му кроці) i-ий процесор переглядає значення ключа $(km + i)$-го запису. В наслідок виконання не більше ніж n кроків шуканий запис буде знайдено, якщо він міститься у файлі. Якщо p_i – ймовірність звертання до i-го запису файла, то математичне сподівання E кількості паралельних порівнянь, необхідних для пошуку запису у файлі, виражається формулою

$$E = \sum_{i=1}^{n} \sum_{j=1}^{m} i p_{(i-1)m+j}\ .$$

Математичне сподівання у випадку різних законів розподілу ймовірностей звертання до записів має наступний вигляд [1, 2]:

- рівномірний розподіл

$$E = \frac{1}{2}\left(\frac{N}{m} + 1\right);$$

- "бінарний" розподіл

$$E = \frac{2^m}{2^m - 1};$$

- закон Зіпфа

$$E = \frac{1}{H_N}\left(H_N + \frac{N}{m} - \frac{1}{2}\ln\frac{N}{m} - C_1\right),$$

де $C_1 = 0.5 \ln 2\pi$;

- узагальнений розподіл

$$E = \frac{1}{H_N^{(c)}}\left[H_N^{(c)} + \frac{N^{1-c}}{1-c}\left(\frac{1-c}{2-c}n - \frac{\alpha^{(c)}(n)}{n^{1-c}}\right)\right],$$

де

$$\alpha^{(c)}(n) = H_n^{(c-1)} - \frac{1}{2-c}n^{2-c}.$$

Зокрема, якщо ймовірності звертання до записів задовільняють закон Зіпфа, то залежність математичного сподівання кількості паралельних порівнянь, необхідних для пошуку запису, від різної кількості процесорів показана на рис. 1.

Рис. 1 – Математичне сподівання кількості паралельних порівнянь, необхідних для пошуку запису, у випадку розподілу ймовірностей звертання до записів за законом Зіпфа, різної кількості процесорів і $N = 10^6$

На підставі порівняння ефективності методу послідовного перегляду (m=1) і методу m-паралельного послідовного перегляду доходимо висновку, що розпаралелювання методу послідовного перегляду веде до підвищення ефективності приблизно в m разів для всіх розглянутих законів розподілу ймовірностей звертання до записів, крім "бінарного".

У випадку **першого варіанту методу *m*-паралельного блочного пошуку** вважаємо, що записи впорядкованого файла розбиті на n блоків по sm записів у кожному і пошук запису у файлі здійснюємо таким чином. Спочатку шукаємо блок, який містить шуканий запис, шляхом перегляду m останніх записів блоків. Після цього пошук продовжуємо у локалізованому блоці за допомогою методу m-паралельного послідовного перегляду. Математичне сподівання E кількості паралельних порівнянь, необхідних для пошуку запису у файлі, запишемо у вигляді суми математичного сподівання кількості паралельних порівнянь, необхідних для локалізації блоку, який містить шуканий запис, і математичне сподівання кількості паралельних порівнянь, необхідних для пошуку запису в локалізованому блоці. Тоді

$$E = \sum_{k=1}^{n}\sum_{i=1}^{s}\sum_{j=1}^{m}(k+i)\,p_{(k-1)ms+(i-1)m+j}\ .$$

Математичне сподівання кількості порівнянь для розглянутих законів розподілу ймовірностей звертання до записів буде мати такий вигляд [3]:

- рівномірний розподіл

$$E = \frac{1}{2}(n+1) + \frac{1}{2}(s+1);$$

- "бінарний" розподіл

$$E = \frac{2^{ms}-s}{2^{ms}-1} + \frac{2^{m}}{2^{m}-1};$$

- закон Зіпфа

$$E = \frac{1}{H_N}\left[2H_N + n + (s-1)\left(\frac{1}{2}\ln n + C_1\right) - \frac{1}{2}\ln(ns) - C_1\right],$$

де $C_1 = 0.5\ln 2\pi$;

- узагальнений розподіл

$$E = \frac{1}{H_N^{(c)}}\left\{2H_N^{(c)} - \frac{N^{1-c}}{1-c}\left[\frac{1-c}{2-c}n - (s-1)\frac{\alpha^{(c)}(n)}{n^{1-c}} + \frac{\alpha^{(c)}(ns)}{(ns)^{1-c}}\right]\right\},$$

де

$$\alpha^{(c)}(n) = H_n^{(c-1)} - \frac{1}{2-c}n^{2-c},$$

$$\alpha^{(c)}(ns) = H_{ns}^{(c-1)} - \frac{1}{2-c}(ns)^{2-c}.$$

Якщо ймовірності звертання до записів задовільняють рівномірний закон розподілу, то залежність математичного сподівання кількості паралельних порівнянь, необхідних для пошуку запису, від різної кількості процесорів показана на рис. 2.

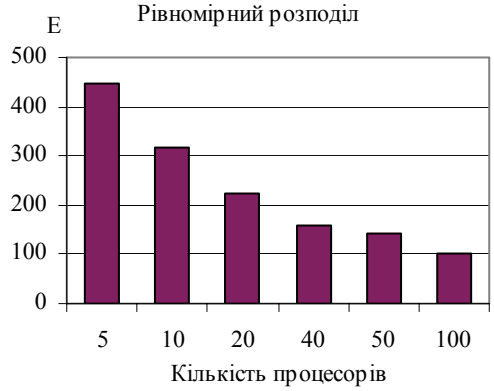

Рис. 2 – Математичне сподівання кількості паралельних порівнянь, необхідних для пошуку запису, у випадку рівномірного закону розподілу ймовірностей звертання до записів і різної кількості процесорів для $N = 10^6$

Порівнюючи ефективність методу блочного пошуку [10] та методу *m*-паралельного блочного пошуку, робимо висновок, що розпаралелювання методу блочного пошуку для всіх розглянутих законів розподілу ймовірностей звертання до записів, окрім "бінарного", суттєво підвищує ефективність. А у випадку "бінарного" закону розподілу ймовірностей збільшення кількості процесорів не підвищує ефективності роботи.

У разі **другого варіанту методу *m*-паралельного блочного пошуку** приймаємо, що записи файла розбиті на *nm* блоків по *sm* записів у кожному, тоді кількість записів у файлі буде $N = snm^2$. Пошук запису у файлі здійснюється наступним чином. Спочатку шукається блок, який містить шуканий запис, використовуючи метод *m*-паралельного послідовного перегляду серед останніх елементів блоків. Після цього пошук продовжується в локалізованому блоці також за допомогою методу *m*-паралельного послідовного перегляду. Математичне сподівання E кількості паралельних порівнянь, необхідних для пошуку запису у файлі, представимо у вигляді суми математичного сподівання кількості паралельних порівнянь, необхідних для локалізації блоку записів, і математичного сподівання кількості паралельних порівнянь, необхідних для пошуку запису в локалізованому блоці:

$$E = \sum_{k=1}^{n}\sum_{l=1}^{m} k\left(\sum_{i=1}^{s}\sum_{j=1}^{m} p_{(k-1)m^2s-(l-1)ms+(i-1)m+j}\right) + \sum_{k=1}^{nm}\sum_{i=1}^{s}\sum_{j=1}^{m} ip_{(k-1)ms+(i-1)m+j}$$

Явний вираз математичного сподівання у випадку різних законів розподілу ймовірностей звертання до записів буде мати вигляд [4]:

- рівномірний розподіл

$$E = \frac{1}{2}(n+1) + \frac{1}{2}(s+1);$$

- "бінарний" розподіл

$$E = \frac{2^{m^2s} - s}{2^{m^2s} - 1} + \frac{2^m}{2^m - 1} - \frac{s}{2^{ms} - 1};$$

- закон Зіпфа

$$E = \frac{1}{H_N}\left[2H_N + n + \frac{1}{2}s\ln(nm) - \frac{1}{2}\ln n - \frac{1}{2}\ln(nms) + C_1(s-2)\right],$$

де $C_1 = 0.5\ln 2\pi$;

- узагальнений розподіл

$$E = \frac{1}{H_N^{(c)}}\left[2H_N^{(c)} + \frac{N^{1-c}}{1-c}\left(\frac{1-c}{2-c}n + \frac{s\alpha^{(c)}(nm)}{(nm)^{1-c}} - \frac{\alpha^{(c)}(n)}{n^{1-c}} - \frac{\alpha^{(c)}(nms)}{(nms)^{1-c}}\right)\right],$$

де

$$\alpha^{(c)}(nm) = H_n^{(c-1)} - \frac{1}{2-c}(nm)^{2-c},$$

$$\alpha^{(c)}(nms) = H_{ns}^{(c-1)} - \frac{1}{2-c}(nms)^{2-c}.$$

Зокрема, якщо ймовірності звертання до записів задовільняють узагальнений закон розподілу, то залежність математичного сподівання кількості паралельних порівнянь, необхідних для пошуку запису, від різної кількості процесорів показана на рис. 3.

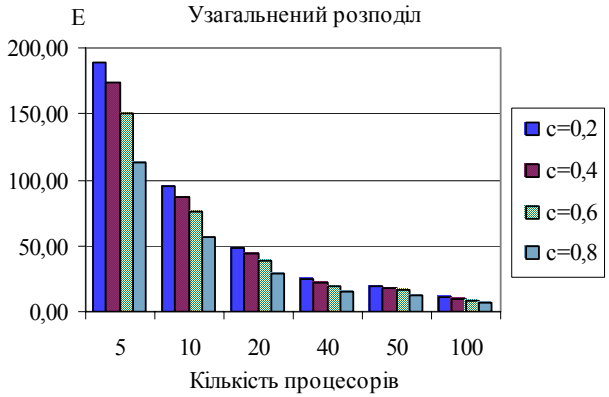

Рис. 3 – Математичне сподівання кількості паралельних порівнянь, необхідних для пошуку запису, у випадку узагальненого закону розподілу ймовірностей звертання до записів і різної кількості процесорів для $N = 10^6$

Якщо порівняти ефективність першого варіанту *m*-паралельного блочного пошуку та другого варіанту *m*-паралельного блочного пошуку, то приходимо до такого висновку: при $m = 1$ перший та другий варіанти методу дають однакову ефективність; але із зростанням кількості процесорів ефективність другого варіанту методу значно зростає порівняно з

ефективністю першого варіанту.

В результаті аналізу методів *m*-паралельного послідовного перегляду, першого та другого варіантів *m*-паралельного блочного пошуку записів у файлах БД приходимо до висновку, що обидва варіанти методу *m*-паралельного блочного пошуку є значно ефективнішими від методу *m*-паралельного послідовного перегляду, для всіх розглянутих законів розподілу ймовірності звертання до записів, крім "бінарного". Також для всіх законів розподілу ймовірностей, крім "бінарного", другий варіант методу *m*-паралельного блочного пошуку є ефективніший, ніж перший. Для розглянутих методів паралельного пошуку в табл. 1 наведені значення математичного сподівання кількості паралельних порівнянь, необхідних для пошуку запису у файлі, який містить $N = 10^6$ записів, у випадку розподілу ймовірностей звертання до записів за законом Зіпфа та різної кількості процесорів.

Таблиця 1. Математичне сподівання кількості паралельних порівнянь, необхідних для пошуку запису у файлі, для розглянутих методів паралельного пошуку у випадку розподілу ймовірностей звертання до записів за законом Зіпфа та різної кількості процесорів

	Паралельні методи пошуку		
m	Послідовний перегляд	1-й варіант блочного пошуку	2-й варіант блочного пошуку
1	69479,99	303,93	303,93
2	34740,25	211,09	152,58
4	17370,39	146,62	76,92
5	13896,42	130,39	61,80
10	6948,49	90,62	31,57
20	3474,54	63,04	16,48
40	1737,57	43,95	8,95
50	1390,18	39,16	7,46
100	695,41	27,43	4,48

2. ОПТИМАЛЬНІ СТРАТЕГІЇ

Використовуючи метод *m*-паралельного послідовного перегляду нами побудовано оптимальні стратегії паралельного пошуку інформації у послідовних упорядкованих файлах баз даних, що зберігається у зовнішній пам'яті ЕОМ, до складу якого входять *m* процесорів, які працюють паралельно і мають спільне поле пам'яті.

Припустимо, що файл, який містить *N* записів, поділений на *n* блоків, у кожному з яких є *ml* записів. Нехай $a_0 = b_0 + d_0 ml$ – час читання блоку записів в основну пам'ять, де b_0, d_0 – деякі сталі; t_0 – час виконання операції *m*-паралельного послідовного перегляду записів в основній пам'яті; p_i – ймовірність звертання до *i*-го запису файла, E_t – математичне сподівання загального часу, необхідного для пошуку запису у файлі. Приймаємо, що для пошуку запису відбувається послідовне читання блоків записів в основну пам'ять і їх *m*-паралельний послідовний перегляд. Тоді

$$E_t = \sum_{k=1}^{n}\sum_{i=1}^{l}\sum_{j=1}^{m}\{ka_0 + [(k-1)l + i]t_0\}\times \times p_{(k-1)ml+(i-1)m+j}.$$

Для E_t знайдено явний вираз у випадку розглянутих законів розподілу ймовірностей звертання до записів і визначено значення параметрів *n* і *l*, за яких математичне сподівання досягає мінімуму [5].

- Рівномірний розподіл

$$E_t = \frac{1}{2}\left[(n+1)\cdot\left(b_0 + \frac{d_0 N}{n}\right) + \left(\frac{N}{m} + 1\right)t_0\right],$$

функція E_t досягає мінімуму при

$$n = (d_0 N / b_0)^{1/2}.$$

- "Бінарний" розподіл

$$E_t = \frac{2^{ml}}{2^{ml}-1}(b_0 + d_0 ml) + \frac{2^m}{2^m - 1}t_0.$$

Для визначення параметра *l*, за якого функція E_t досягає мінімуму, отримаємо рівняння

$$2^{ml} = 1 + \left(ml + \frac{b_0}{d_0}\right)\ln 2.$$

- Закон Зіпфа

$$E_t = \frac{1}{H_N}\left[\left(H_N + n - \frac{1}{2}\ln n - C_1\right)\left(b_0 + \frac{d_0 N}{n}\right) + \left(H_N + \frac{N}{m} - \frac{1}{2}\ln\frac{N}{m} - C_1\right)t_0\right],$$

де $C_1 = 0.5\ln 2\pi$;

Для наближеного обчислення значення параметра n, за якого E_t досягає мінімуму, маємо рівняння

$$2n^2 - n = \frac{d_0 N}{b_0}\left(2H_N + 1 - \ln n - 2C_1\right).$$

- узагальнений розподіл

$$E_t = \frac{1}{H_N^{(c)}}\left\{\left[H_N^{(c)} - \frac{N^{1-c}}{1-c}\left(\frac{c-1}{2-c}n + \frac{\alpha^{(c)}(n)}{n^{1-c}}\right)\right]\left(b_0 + \frac{d_0 N}{n}\right) + \left[H_N^{(c)} + \frac{N^{1-c}}{1-c}\left(\frac{c-1}{2-c}\frac{N}{m} + \frac{\alpha^{(c)}(N/m)}{(N/m)^{1-c}}\right)\right]t_0\right\},$$

де

$$\alpha^{(c)}(n) = H_n^{(c-1)} - \frac{1}{2-c}n^{2-c}.$$

Параметр n, за якого функція E_t досягає мінімуму, можемо визначити з наступного рівняння:

$$n^{3-c} + (2-c)\cdot\left(n + \frac{2-c}{1-c}\frac{d_0}{b_0}N\right)\alpha^{(c)}(n) = (2-c)\frac{d_0}{b_0}N^c n^{1-c}H_N^{(c)} + \frac{2-c}{1-c}n\left(n + \frac{d_0}{b_0}N\right)\left(\alpha^{(c)}(n+1) - \alpha^{(c)}(n)\right).$$

Зокрема, якщо ймовірності звертання до записів задовільняють узагальнений закон розподілу, $N = 10^6$, $b_0/d_0 = 20$, то значення параметра l зображені в табл. 2.

На відміну від параметра l, оптимальні значення параметра n, за яких математичне сподівання загального часу, необхідного для пошуку запису у файлі, досягає мінімуму, не залежать від кількості процесорів, а залежать лише від зміни закону розподілу ймовірностей звертання до записів для всіх законів розподілу, крім "бінарного". Зокрема, залежність оптимального значення параметра n від зміни закону розподілу ймовірностей звертання до записів для $N = 10^6$ і $b_0/d_0 = 20$ зображено відповідно на рис. 4.

Таблиця 2. Оптимальні значення параметра *l* у випадку узагальненого закону розподілу ймовірностей та різної кількості процесорів

m	c=0.2	c=0.4	c=0.6	c=0.8
1	4201,68	3831,42	3278,69	2421,31
2	2100,84	1915,71	1639,34	1210,65
4	1050,42	957,85	819,67	605,33
5	840,34	766,28	655,74	484,26
10	420,17	383,14	327,87	242,13
20	210,08	191,57	163,93	121,07
40	105,04	95,79	81,97	60,53
50	84,03	76,63	65,57	48,43
100	42,02	38,31	32,79	24,21

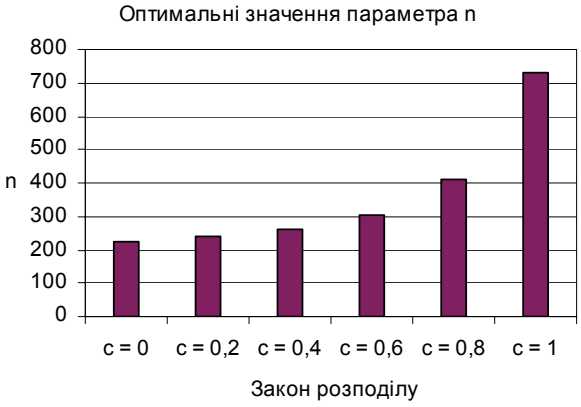

Рис. 4 – Оптимальні значення параметра n у випадку $N = 10^6$ та $b_0/d_0 = 20$ для таких законів розподілу ймовірностей звертання до записів: рівномірного (c=0), узагальненого та Зіпфа (c=1)

3. ВИСНОВКИ

Розглянуто метод m-паралельного послідовного перегляду та два варіанти методу m-паралельного блочного пошуку. Досліджено ефективність цих методів для таких законів розподілу ймовірностей звертання до записів як: рівномірний, "бінарний", Зіпфа й узагальнений, частковим випадком якого є розподіл, що наближено задовольняє правило "80-20". За критерій ефективності взято математичне сподівання кількості паралельних порівнянь, необхідних для пошуку запису у файлі. Проведено порівняльний аналіз ефективності методів і для кожного розглянутого закону розподілу ймовірностей звертання до записів визначено свій найкращий метод.

Побудовано оптимальні стратегії пошуку записів в послідовних файлах, які зберігаються у зовнішній пам'яті багатопроцесорної ЕОМ, для різних законів розподілу ймовірностей звертання до записів. За критерій ефективності взято

математичне сподівання загального часу, необхідного для пошуку запису у файлі. Визначені значення параметрів, за яких математичне сподівання досягає мінімуму.

СПИСОК ЛІТЕРАТУРИ

[1] Лісовець В. Я., Цегелик Г. Г. Метод m-паралельного послідовного перегляду записів та його використання для пошуку інформації у послідовних файлах баз даних // Фізико-математичне моделювання та інформаційні технології. – 2007. – Вип. 5. — С. 109-119.

[2] Лісовець В., Цегелик Г. Метод m-паралельного послідовного пошуку записів у файлах баз даних і його ефективність // Вісн. Львів. ун-ту. Сер. прикл. матем. та інформ. –2006. – Вип. 13.– С. 177-186.

[3] Лісовець В. Я., Цегелик Г. Г. Метод m-паралельного блочного пошуку записів у файлах баз даних та його ефективність // Відбір та обробка інформації. – 2007. – Вип. 27(103). – С. 87-92.

[4] Лісовець В. Я., Цегелик Г. Г. Один з варіантів методу m-паралельного блочного пошуку записів і його ефективність // Фізико-математичне моделювання та інформаційні технології. – 2008. – Вип. 7. – С. 103-111.

[5] Лісовець В., Цегелик Г. Моделювання та оптимізація паралельного пошуку інформації у файлах баз даних // Матеріали третьої міжнародної науково-технічна конференції: "Комп'ютерні науки та інформаційні технології" CSIT'2008 (25-27 вересня 2008р.). – Львів: Видавництво ПП "Вежа і Ко", 2008, С. 277-280

[6] Кнут Д. Искусство программирования для ЭВМ. Т. 3: Сортировка и поиск. – М.: Изд. дом "Вильямс", 2000. – 832 с.

[7] Мартин Дж. Организация баз данных в вычислительных системах. – М: Мир, 1980. – 644 с.

[8] Цегелик Г. Г. Организация и поиск информации в базах данных. – Львов: Вища шк., 1987. – 176 с.

[9] Цегелик Г.Г. Системы распределенных баз данных. – Львов: Світ, 1990, – 168с.

[10] Мельничин А. В. Моделювання та оптимізація доступу до інформації файлів баз даних. Автореферат дисертації на здобуття наукового ступеня кандидата технічних наук: спеціальність 01.05.03 "Математичне та програмне забезпечення обчислювальних машин і систем" / Мельничин А.В. – Львів: "Львівська політехніка", 2009, – 20с.

***Лісовець Володимир Ярославович**, народився 29 липня 1983 року в м. Дрогобичі Львівської області.*

В 2000 – 2005 рр навчався в Львівському національному університет (ЛНУ) ім. І. Франка на факультеті прикладної математики та інформатики. В 2005р. отримав диплом магістра.

З листопада 2007р аспірант кафедри математичного-моделювання соціально-економічних процесів факультету прикладної математики та інформатики ЛНУ ім. Франка.

***Цегелик Григорій Григорович**, народився 1942р. Закінчив Львівський державний університет, кандидат фіз.-мат наук з 1969р. Доктор фіз.-мат наук з 1990р. З 1991р. професор. З 1974р по 1984р очолював кафедру механізованої обробки економічної інформації. З 1993р по 2000р очолював кафедру обчислювальної математики, а з 2000р по теперішній час очолює кафедру математичного моделювання соціально-економічних процесів факультету прикладної математики та інформатики Львівського національного університету ім. І.Франка. Наукові інтереси: теоретичні основи інформатики та кібернетики, чисельні методи аналізу, математичне моделювання економічних процесів. Автор та співавтор понад 500 публікацій, у тому числі 3-ох індивідуальних монографій, 1-го підручника та 2-ох посібників з грифом МОН України.*

computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

MODELING AND OPTIMIZATION OF PARALLEL INFORMATION SEARCHING IN FILES

Volodymyr Lisovets, Hryhoriy Tsehelyk

Ivan Franko National University of L'viv,
1, Universytetska str., Lviv, 79000, Ukraine,
kafmmsep@franko.lviv.ua

Abstract: *In this article the m-parallel method of sequential field searching and two variants of m-parallel block field searching method are offered. These methods are oriented to be used in multiprocessing system for information searching in files of database. We research the effectiveness of these methods for different probability distribution law of field access. The mathematical expectation of number of parallel comparisons necessary for field searching in files is taken as a criterion of effectiveness. The effectiveness of the methods is compared and analyzed. The best of offered methods is founded for every considered probability distribution. Optimal strategies of field searching in sequenced files stored in external memory of multiprocessing system are made. In this case the mathematical expectation of total time needed for field searching in files is taken as a criterion of effectiveness.*

Keywords: *multiprocessing system, mathematical modeling, parallel searching, database.*

1. THE METHODS OF PARALLEL SEARCHING AND THEIR EFFECTIVENESS

The following methods of parallel field searching in files of database for multiprocessing system are considered [1-4]:

- method of *m*-parallel sequential field searching;
- the first variant of *m*-parallel block field searching method;
- the second variant of *m*-parallel block field searching method.

We analyze the effectiveness of these methods for different probability distribution law of field access (discrete uniform, binomial, Zipf and generalized the partial occasion of witch is the probability distribution approximately satisfying the rule "80 – 20" [5-7]). The mathematical expectation of number of parallel comparisons necessary for field searching in files is taken as a criterion of effectiveness.

Let's consider the method of *m*-parallel sequential field searching.

Suppose that multiprocessing system consists of *m* processors working parallel and having common memory. Let's enumerate the processors by natural numbers from 1 to *m*. The main point of the method of *m*-parallel sequential field searching is the following. Divide conventionally all fields of file into blocks and each of them includes *m* fields. Let $N = n \cdot m$ is the number of fields in file, where *n* is the number of blocks. When method of *m*-parallel sequential field searching is used the field searching will consist of the next steps. On the first step the processor number *i* searches the value of the key of field number *i*. In this case the process of searching can be successful or failed. Each processor must exchange data with other processors on every step. In case of successful searching the field searching process is finished. In case of failed searching the processor number *i* searches the value of the key of filed number $(m + i)$ on the second step etc. If the field searching is failed on the step number *k* then on the step number $(k + 1)$ the processor number *i* searches the value of the key of filed number $(km + i)$. If the required field is located in the file then it will be found after *n* steps. If p_i is probability of access to field number *i* then mathematical expectation *E* of number of parallel comparisons necessary for field searching in files is calculated by following formula

$$E = \sum_{i=1}^{n} \sum_{j=1}^{m} i p_{(i-1)m+j} \cdot$$

In case of using the first variant of *m*-parallel block field searching method we suppose that the fields of ordered file are divided into *n* blocks each

of them includes *sm* fields. Then the filed searching will be done in this way. First of all we search the block including the needed field by looking the last *m* fields of file. After that we continue the searching in the located block by the method of *m*-parallel sequential field searching. The mathematical expectation *E* of number of parallel comparisons necessary for field searching in files lets write as the sum of mathematical expectation of number of parallel comparisons necessary for block location which includes the needed field and mathematical expectation of number of parallel comparisons necessary for field searching in the located block. Then

$$E = \sum_{k=1}^{n}\sum_{i=1}^{s}\sum_{j=1}^{m}(k+i)\,p_{(k-1)ms+(i-1)m+j}\ .$$

In case of using the second variant of *m*-parallel block field searching method let suppose that the fields of ordered file are divided into *nm* blocks and each of them includes *sm* fields. Then the number of fields of file will be $N = snm^2$. Then the field searching will be done in the following way. First of all we search the block including the needed field by using the method of *m*-parallel sequential searching among the last elements of blocks. After that we continue the searching in the located block by the method of *m*-parallel sequential searching. The mathematical expectation *E* of number of parallel comparisons necessary for field searching in files lets write as the sum of mathematical expectation of number of parallel comparisons necessary for block location which includes the needed field and mathematical expectation of number of parallel comparisons necessary for field searching in the located block:

$$E = \sum_{k=1}^{n}\sum_{l=1}^{m} k\left(\sum_{i=1}^{s}\sum_{j=1}^{m} p_{(k-1)m^2s-(l-1)ms+(i-1)m+j}\right) + \sum_{k=1}^{nm}\sum_{i=1}^{s}\sum_{j=1}^{m} ip_{(k-1)ms+(i-1)m+j}$$

2. THE OPTIMAL STRATEGIES

Using the method of *m*-parallel sequential field searching we create the optimal strategies of parallel information searching in sequenced ordered files of database that are stored in external memory of multiprocessing system. Suppose that multiprocessing system consists of *m* processors working parallel and having common memory.

Suppose that file including *N* fields is divided into *n* blocks and each of them includes *ml* fields. Let $a_0 = b_0 + d_0ml$ is the time of block field reading in the RAM, where b_0, d_0 are some constants; t_0 is the one step time of the method of m-parallel sequential field searching in the RAM; p_i is the probability of access to field number *i*, E_t is the mathematical expectation of total time needed for field searching in file. Let suppose that first of all the field blocks are read to RAM from external memory and then *m*-parallel sequential searching method is used. Then

$$E_t = \sum_{k=1}^{n}\sum_{i=1}^{l}\sum_{j=1}^{m}\{ka_0 + [(k-1)l + i]t_0\}\times \times p_{(k-1)ml+(i-1)m+j}\ .$$

3. CONCLUSION

The *m*-parallel method of sequential field searching and two variants of *m*-parallel block field searching method are considered. We researched the effectiveness of these methods for different probability distribution law of field access (discrete uniform, binomial, Zipf and generalized the partial occasion of witch is the probability distribution approximately satisfying the rule “80 – 20”). The mathematical expectation of number of parallel comparisons necessary for field searching in files was taken as a criterion of effectiveness. The effectiveness of the methods was compared and analyzed. The best of offered methods was founded for every considered probability distribution.

Optimal strategies of field searching in sequenced files stored in external memory of multiprocessing system were made for different probability distribution law of field access. In this case the mathematical expectation of total time needed for field searching in files was taken as a criterion of effectiveness. The values of parameters when mathematical expectation reaches the minimum are founded.