



УКРАЇНА

(19) **UA** (11) **93797** (13) **U**
(51) МПК (2014.01)
G06F 7/00

ДЕРЖАВНА СЛУЖБА
ІНТЕЛЕКТУАЛЬНОЇ
ВЛАСНОСТІ
УКРАЇНИ

(12) ОПИС ДО ПАТЕНТУ НА КОРИСНУ МОДЕЛЬ

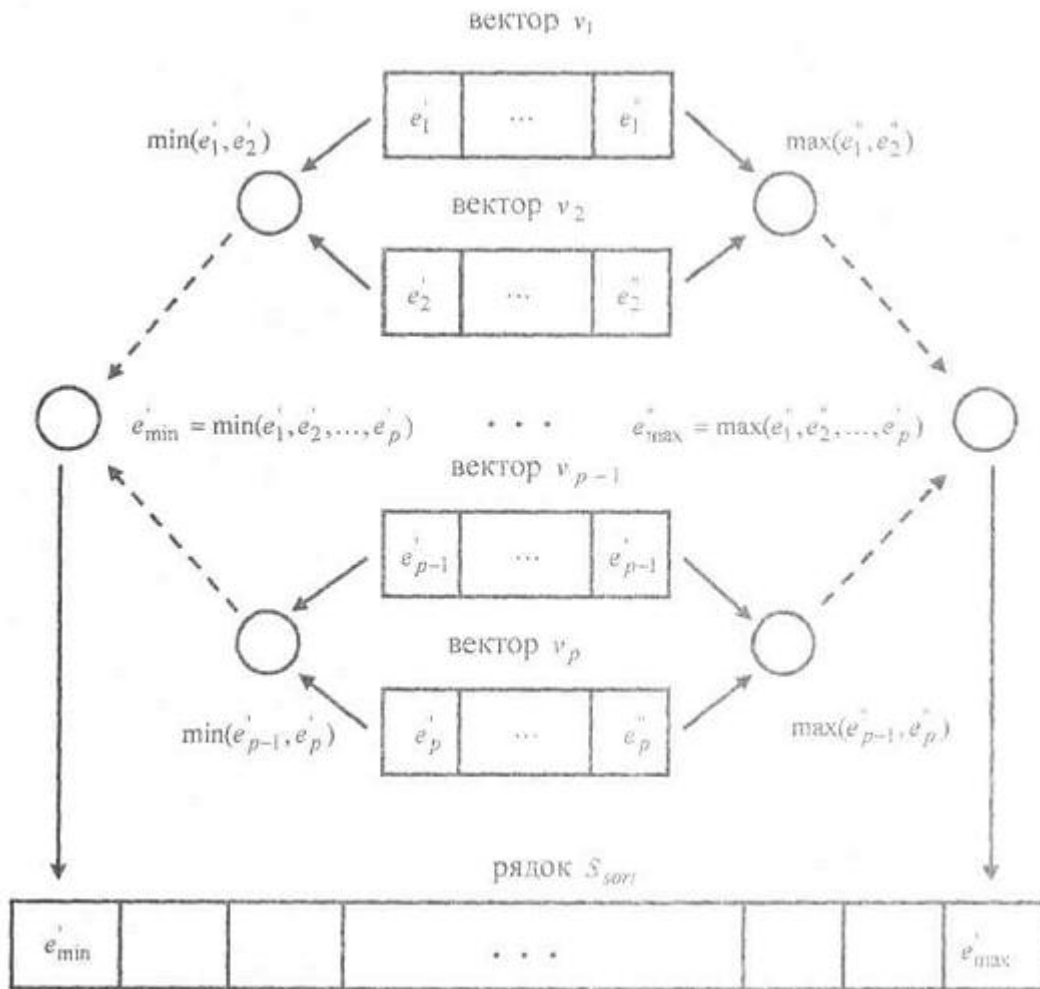
(21) Номер заявки: u 2014 05831	(72) Винахідник(и): Семеренко Василь Петрович (UA), Матрос Дмитро Олександрович (UA)
(22) Дата подання заявки: 29.05.2014	(73) Власник(и): ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ, Хмельницьке шосе, 95, м. Вінниця, 21021 (UA)
(24) Дата, з якої є чинними права на корисну модель: 10.10.2014	
(46) Публікація відомостей про видачу патенту: 10.10.2014, Бюл.№ 19	

(54) СПОСІБ ДЛЯ ПАРАЛЕЛЬНОГО ЗОВНІШНЬОГО СОРТУВАННЯ ДАНИХ

(57) Реферат:

Спосіб для паралельного зовнішнього сортування даних складається із етапу проміжного сортування, під час якого невідсортований рядок S із n чисел поділяється на p векторів із m чисел кожний, які сортуються паралельно, і етапу повного сортування, під час якого на основі відсортованих векторів з використанням двох паралельних процесів методом вибору за допомогою бінарного дерева формується відсортований рядок відсортовані вектори методом злиття формують відсортований рядок S_{sort} із заданих n чисел на k -й ітерації етапу повного сортування із чергових мінімальних значень чисел всіх векторів з використанням першого процесу визначається найменше число, яке переміщається в k -у позицію рядка S_{sort} і одночасно із чергових максимальних значень всіх векторів з використанням другого процесу, визначається найбільше число, яке переміщається в $(n-k+1)$ -у позицію рядка S_{sort} , тривалість ітерації тієї частини етапу повного сортування, яка суміщається із етапом проміжного сортування, повинна бути такою ж, як і тривалість ітерації етапу проміжного сортування ($k = 1 \div \frac{n}{2}$).

UA 93797 U



Фиг. 1

Корисна модель належить до операцій сортування даних і може бути використана в системах управління базами даних, в інформаційно-вимірювальних та інших системах, пов'язаних з обробкою великих обсягів даних.

Відомі способи зовнішнього сортування даних: прямого злиття, багатофазного злиття, каскадного злиття [Д. Э. Кнут Искусство программирования для ЭВМ, т. 3, Сортировка и поиск. Пер. с англ. - М.: Мир, 1978. - С. 174]. Недоліками цих способів є велика кількість операцій запису і зчитування для зовнішніх носіїв даних, наявність спеціальних умов для ефективного виконання задачі сортування.

Найбільш близьким за технічною суттю до запропонованого є спосіб паралельного злиття/сортування за допомогою паралельних процесорів [патент США US 2005/0144167 A1 М. кл., G06F 7/00, Jun. 30, 2005], який складається із етапу проміжного сортування і етапу повного сортування, на етапі проміжного сортування невідсортований рядок S із n чисел поділяється на P векторів із m чисел кожний, які сортуються паралельно, на етапі повного сортування відсортовані вектори методом злиття формують відсортований рядок чисел більшої довжини одночасно в порядку зростання значень чисел і в порядку спадання значень чисел, етап повного сортування розпочинається: закінчення етапу проміжного сортування і триває $\log_2 P$ ітерацій, на i -й ітерації із P відсортованих векторів методом попарного злиття кожних двох сусідніх рядків (двох сусідніх векторів на першій ітерації) із $\frac{P}{2^{i-1}}$ чисел формується відсортований рядок із $\frac{P}{2^i}$ чисел з використанням 2^{i-1} паралельних процесів ($i = 1 \div \log_2 P$).

Недоліками цього способу є великі витрати часу внаслідок великої кількості операцій запису і зчитування для зовнішніх носіїв даних та послідовного використання двох процедур сортування: процедури сортування частин рядків і багатокрокової процедури злиття частин рядка в один відсортований рядок.

В основу корисної моделі поставлена задача створення способу для паралельного зовнішнього сортування даних, згідно якого зменшується кількість операцій запису та зчитування даних на зовнішній носій даних та суміщаються в часі процеси сортування частин рядків (векторів) і злиття відсортованих векторів в один відсортований рядок, в результаті чого зменшуються загальні витрати часу на виконання зовнішнього сортування даних.

Поставлена задача вирішується тим, що в способі для паралельного зовнішнього сортування даних, який складається із етапу проміжного сортування і етапу повного сортування, на етапі проміжного сортування невідсортований рядок S із n чисел поділяється на P векторів із m чисел кожний, які сортуються паралельно, на етапі повного сортування відсортовані вектори методом злиття формують відсортований рядок чисел більшої довжини одночасно в порядку зростання значень чисел і в порядку спадання значень чисел, причому на етапі проміжного сортування P векторів протягом $\frac{m}{2}$ ітерацій сортуються з використанням $2P$ паралельних процесів методом вибору за допомогою бінарного дерева, на одній ітерації етапу проміжного сортування процес Z_i^h сортує вектор V_i в порядку зростання і одночасно з ним процес Z_i^{ls} сортує вектор V_i в порядку спадання і визначаються відповідні мінімальні і максимальні значення вектора V_i ($i = 1 \div P$), після закінчення першої ітерації етапу проміжного сортування розпочинається етап повного сортування, на якому протягом $\frac{n}{2}$ ітерацій з використанням паралельних процесів Z_{end}^h та Z_{end}^{ls} методом вибору за допомогою бінарного дерева формується відсортований рядок S_{sort} із заданих n чисел, на k -й ітерації етапу повного сортування із чергових мінімальних значень чисел всіх векторів з використанням першого процесу Z_{end}^h визначається найменше число, яке переміщається в k -у позицію рядка S_{sort} і одночасно із чергових максимальних значень чисел всіх векторів з використанням процесу Z_{end}^{ls} визначається найбільше число, яке переміщається в $(n-k+1)$ -у позицію рядка S_{sort} ($k = 1 \div \frac{n}{2}$), після вибору мінімального (максимального) числа із відповідного вектора на його місце переміщається чергове в порядку зростання (спадання) число цього вектора, тривалість ітерації тієї частини етапу повного сортування, яка суміщається із етапом проміжного сортування, повинна бути такою ж, як і тривалість ітерації етапу проміжного сортування.

Спосіб здійснюється наступним чином. Задано невідсортований рядок із n чисел $S = \{s_1, s_2, \dots, s_n\}$, з якого необхідно сформувати рядок чисел S_{sort} з відсортованими в порядку зростання числами. Рядок S поділяється на p векторів v_1, v_2, \dots, v_p , кожний з яких містить по m чисел.

5 Спосіб сортування складається з двох етапів:

- етапу проміжного сортування;
- етапу повного сортування.

Етап проміжного сортування полягає у формуванні p відсортованих векторів за допомогою $2p$ паралельних процесів, створених відповідно у $2p$ процесорах. Процес Z_i^h сортує вектор v_i в

10 порядку зростання і одночасно з ним процес Z_i^s сортує вектор v_i в порядку спадання ($i = 1 \div p$). Для сортування векторів використовується метод сортування вибором за допомогою бінарного дерева. Для вектора v_i формується дерево D_i^{\max} та дерево D_i^{\min} ($i = 1 \div p$). Числа вектора v_i утворюють кінцеві вершини обох дерев.

Протягом першої ітерації визначаються мінімальне і максимальне числа кожного вектора.

15 На першому кроці першої ітерації сортування вектора v_i порівнюються сусідні числа і визначаються з них мінімальне число e_i' (у дереві D_i^{\min}) і максимальне число e_i'' (у дереві D_i^{\max}). На j -му рівні обох дерев порівнюються відповідні пари чисел, знайдених на $(j-1)$ -му рівні ($j = 1 \div \log_2 m$). В кінці першої ітерації в корні дерева D_i^{\min} формується мінімальне число e_{\min}' вектора v_i , яке поміщається на початок вектора v_i , а в корні дерева D_i^{\max} формується

20 максимальне число e_{\max}'' вектора v_i , яке поміщається в кінець вектора v_i (Фіг. 1).

На наступних ітераціях визначають чергові мінімальні і максимальні числа кожного вектора серед невідсортованих чисел цих векторів. Через $\frac{m}{2}$ ітерацій буде завершено сортування всіх векторів. Після завершення етапу проміжного сортування в початкових позиціях всіх векторів знаходяться мінімальні числа, а в кінцевих позиціях - максимальні числа відповідних векторів.

25 Кількість кроків для виконання етапу проміжного сортування складає не більше, ніж N_1 кроків:

$$N_1 = \frac{m}{2} \log_2 m$$

Етап повного сортування полягає у сортуванні масиву M через злиття відсортованих векторів v_i . Для виконання цього етапу використовується метод сортування вибором за

30 допомогою бінарних дерев D_{end}^{\max} і D_{end}^{\min} та паралельних процесів Z_{end}^h і Z_{end}^s . Початкові позиції векторів v_i утворюють вектор мінімальних значень v_{\min} , а кінцеві позиції векторів v_i утворюють вектор максимальних значень v_{\max} . Числа вектора v_{\min} утворюють кінцеві вершини дерева D_{end}^{\min} , а числа вектора v_{\max} утворюють кінцеві вершини дерева D_{end}^{\max} .

35 В кінці першої ітерації другого етапу сортування в корні дерева D_{end}^{\min} за допомогою процесу Z_{end}^h формується найменше число вектора v_{\min} , яке поміщається на початок рядка S_{sort} , а в корні дерева D_{end}^{\max} за допомогою процесу Z_{end}^s формується найбільше число вектора v_{\max} , яке поміщається в кінець рядка S_{sort} .

40 На k -й ітерації етапу повного сортування із вектора мінімальних значень v_{\min} з використанням першого процесу Z_{end}^h визначається найменше число, яке переміщується в k -у позицію рядка S_{sort} і одночасно із вектора максимальних значень v_{\max} з використанням процесу Z_{end}^s визначається найбільше число, яке переміщується в $(n - k + 1)$ -у позицію рядка S_{sort} ($k = 1 \div \frac{n}{2}$).

Числа, які поміщаються у відсортований рядок S_{sort} , вилучаються із векторів v_{\min} та v_{\max} , а на їх місце поступають із відповідних векторів v_i наступні по зростанню та спаданню числа.

Через $\frac{n}{2}$ ітерацій буде завершено формування відсортованого масиву M_{sort} . Кількість кроків для виконання етапу повного сортування складає не більше, ніж N_2 кроків:

$$N_2 = \frac{n}{2} \log_2 p$$

Для скорочення загального часу сортування етап проміжного та етап повного сортування можна частково сумістити і виконувати паралельно. Можливість суміщення в часі двох етапів сортування основана на тому, що після кожної ітерації етапу проміжного сортування векторів підготовляються початкові дані для проведення однієї ітерації етапу повного сортування. Відразу після закінчення першої ітерації етапу проміжного сортування можна розпочинати етап повного сортування. Етап проміжного сортування триває перші $t_1 = \frac{m}{2}$ ітерацій, етап повного

10 сортування триває з другої по t_2 ітерацію ($t_2 = \frac{n}{2} + 1$) (Фіг.2).

Протягом того періоду часу, коли обидва етапи сортування взаємно суміщаються, тривалість ітерацій на цих етапах повинна бути однаковою і визначатися тривалістю тієї ітерації, що вимагає більше часу.

15 Порівняльна оцінка витрат часу на проведення сортування запропонованим та відомим способами.

Протягом однієї ітерації першого етапу сортування для кожного процесу відбувається m операцій зчитування елементів вектора з оперативної пам'яті, $\log_2 m$ операцій сортування процесором та дві операції запису результату в оперативну пам'ять (обмін позиціями для мінімального або максимального чисел), отже загальна тривалість ітерації складає

20
$$T_s^{1,it} = m \times \tau_{\text{RAM}} + \log_2 m \times \tau_{\text{proc}} + 2\tau_{\text{RAM}}$$

де τ_{RAM} - час доступу до оперативної пам'яті,

τ_{proc} - час роботи процесора.

Загальна тривалість першого етапу сортування запропонованим способом складає

$$T_s^1 = \frac{m}{2} T_s^{1,it}$$

25 Тривалість перших етапів запропонованого і відомого способів приблизно однакові.

Протягом однієї ітерації другого етапу сортування запропонованим способом для кожного процесу відбувається m операцій зчитування елементів векторів з оперативної пам'яті, $\log_2 p$ операцій сортування процесором та одна операція запису результату на зовнішній носій даних (наприклад, магнітний диск), отже загальна тривалість ітерації складає

30
$$T_s^{2,it} = m \times \tau_{\text{RAM}} + \log_2 m \times \tau_{\text{proc}} + \tau_{\text{disk}}$$

де τ_{disk} - час доступу до зовнішнього носія даних.

Загальна тривалість другого етапу сортування запропонованим способом складає

$$T_s^2 = \frac{n}{2} \times T_s^{2,it}$$

35 Протягом однієї ітерації другого етапу сортування відомим способом під час злиття на зовнішньому носії даних двох рядків із k чисел в один рядок із $2k$ чисел відбувається k операцій зчитування із зовнішнього носія даних, $2k$ операцій запису на зовнішній носій даних, а також k операцій порівняння за допомогою процесора ($k = 2m, 4m, \dots$), отже загальна тривалість ітерації складає

$$T_k^{2,it} = \frac{1}{2} (2n \times \tau_{\text{disk}} + 2 \times \tau_{\text{proc}})$$

40 Загальна тривалість другого етапу сортування відомим способом складає

$$T_k^2 = \frac{1}{2} (2n \times \log_2 p - 1) \times \tau_{\text{disk}} + \frac{n}{2} \times \tau_{\text{disk}} + \frac{n}{2} \times \log_2 p \times \tau_{\text{proc}}$$

При виконанні процедури зовнішнього сортування на сучасних комп'ютерах основні витрати часу припадають на звертання до зовнішніх носіїв даних ($\tau_{\text{disk}} \gg \tau_{\text{RAM}}, \tau_{\text{disk}} \gg \tau_{\text{proc}}$). Тому можна вважати, що

45
$$T_s^2 \approx \frac{n}{2} \times \tau_{\text{disk}}, T_k^2 = \frac{1}{2} (2n \times \log_2 p - 1) \times \tau_{\text{disk}} + \frac{n}{2} \times \tau_{\text{disk}}$$

Таким чином, в η разів зменшуються витрати часу на зовнішнє сортування запропонованим і відомим способами за рахунок зменшення кількості операцій звертання до зовнішніх носіїв даних:

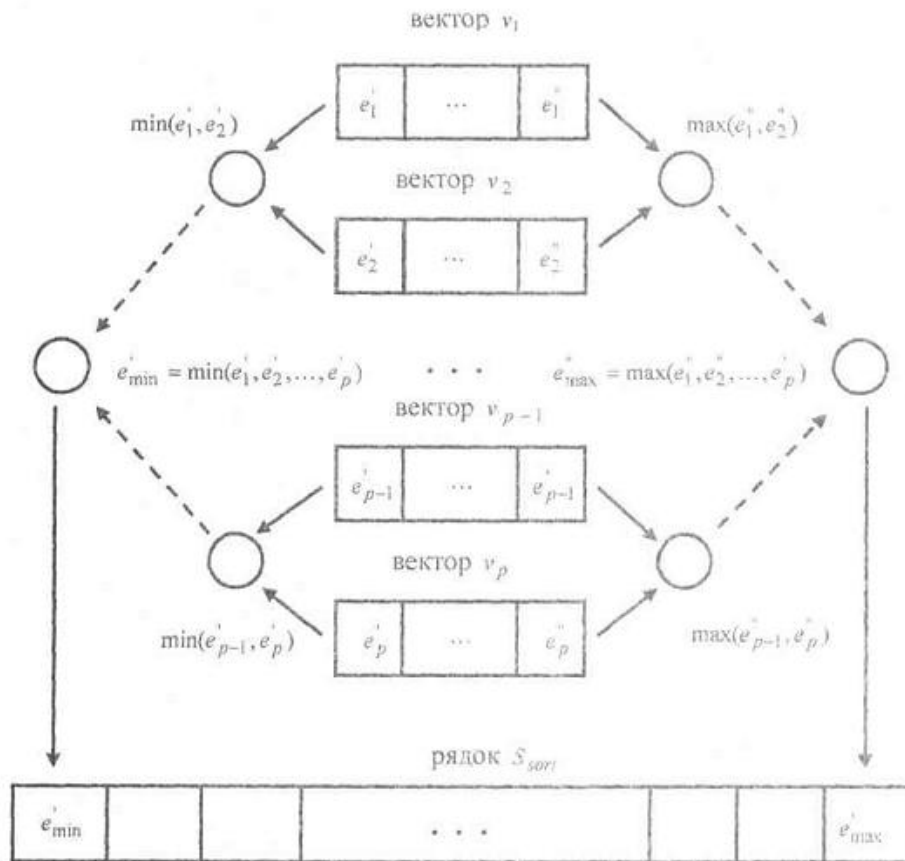
$$\eta = \frac{T_k^2}{T_s^2} \approx 2(\log_2 p - 1) + 1.$$

5

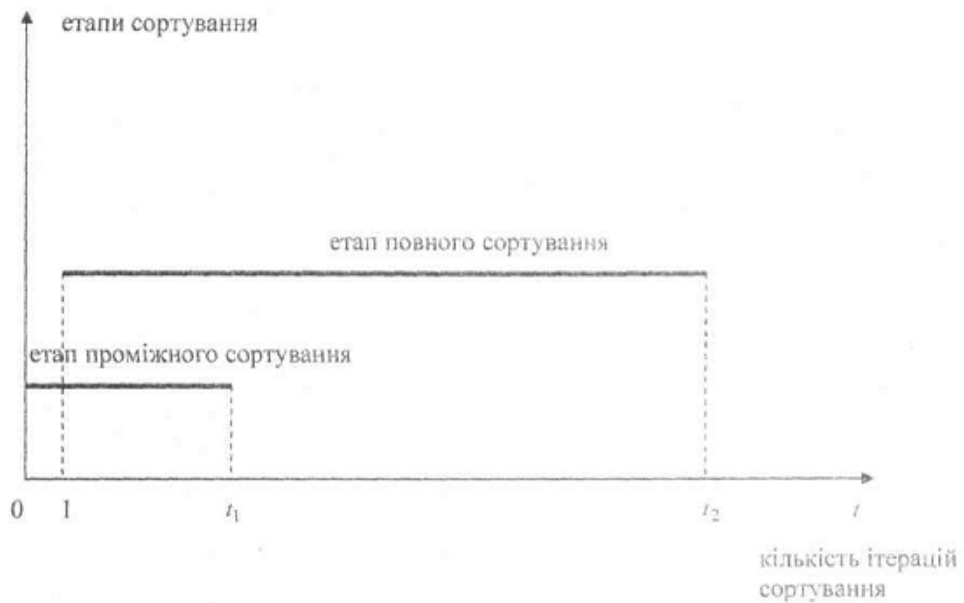
ФОРМУЛА КОРИСНОЇ МОДЕЛІ

Спосіб для паралельного зовнішнього сортування даних, який складається із етапу проміжного сортування і етапу повного сортування, на етапі проміжного сортування невідсортований рядок S із n чисел поділяється на p векторів із m чисел кожний, які сортуються паралельно, на етапі повного сортування відсортовані вектори методом злиття формують відсортований рядок чисел більшої довжини одночасно в порядку зростання значень чисел і в порядку спадання значень чисел, який **відрізняється** тим, що на етапі проміжного сортування p векторів протягом $\frac{m}{2}$ ітерацій сортуються з використанням $2p$ паралельних процесів методом вибору за допомогою бінарного дерева, на одній ітерації етапу проміжного сортування процес Z_i^h сортує вектор v_i в порядку зростання і одночасно з ним процес Z_i^s сортує вектор v_i - в порядку спадання і визначаються відповідні мінімальні і максимальні значення вектора $v_i (= 1 \div p)$, після закінчення першої ітерації етапу проміжного сортування розпочинається етап повного сортування, на якому протягом $\frac{n}{2}$ ітерацій з використанням паралельних процесів Z_{end}^h та Z_{end}^s методом вибору за допомогою бінарного дерева формується відсортований рядок S_{sort} із заданих n чисел, на k -й ітерації етапу повного сортування із чергових мінімальних значень чисел всіх векторів з використанням першого процесу Z_{end}^h визначається найменше число, яке переміщається в k -у позицію рядка S_{sort} і одночасно із чергових максимальних значень чисел всіх векторів з використанням процесу, визначається найбільше число, яке переміщається в $(n - k + 1)$ -у позицію рядка $S_{sort} (k = 1 \div \frac{n}{2})$, після вибору мінімального (максимального) числа із відповідного вектора на його місце переміщається чергове в порядку зростання (спадання) число цього вектора, тривалість ітерації тієї частини етапу повного сортування, яка суміщається із етапом проміжного сортування, повинна бути такою ж, як і тривалість ітерації етапу проміжного сортування.

30



Фіг. 1



Фіг. 2

Комп'ютерна верстка С. Чулій

Державна служба інтелектуальної власності України, вул. Урицького, 45, м. Київ, МСП, 03680, Україна

ДП "Український інститут промислової власності", вул. Глазунова, 1, м. Київ – 42, 01601