УДК 681.3.06

В.А. ЛИТВИНОВ, С.Я. МАЙСТРЕНКО, В.И. ХОДАК

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ АЛГОРИТМА ПРОИЗВОЛЬНО-ПОСЛЕДОВАТЕЛЬНОЙ ОБРАБОТКИ ФАЙЛОВ В ОДНОРОДНОЙ ЗАПОМИНАЮЩЕЙ СРЕДЕ

Abstract: *It is described the advanced algorithm of the direct-sequential processing of an ordered file. The results of the imitation modeling the algorithm and comparative estimations of its speed of response in homogeneous memory are given.*
Key words: *file processing, key search, imitation modeling.*

Анотація: *Описується удосконалений алгоритм довільно-послідовної обробки упорядкованого файла. Приводяться результати імітаційного моделювання алгоритму і порівняльні оцінки його швидкодії в однорідному запам'ятовуючому середовищі.*
Ключові слова: *обробка файлів, пошук по ключу, імітаційне моделювання.*

Аннотация: *Описывается усовершенствованный алгоритм произвольно-последовательной обработки упорядоченного файла. Приводятся результаты имитационного моделирования алгоритма и сравнительные оценки его быстродействия в однородной запоминающей среде.*
Ключевые слова: *обработка файлов, поиск по ключу, имитационное моделирование.*

1. Введение

Для повышения эффективности (скорости) обработки последовательного упорядоченного файла, расположенного во внешней памяти, в [1] был предложен алгоритм произвольно-последовательной обработки (ППО), значительно ускоряющий процесс поиска-выборки элементов файла по сравнению как с дихотомическим поиском, так и чисто последовательной обработкой. Сокращение времени поиска-выборки в рассматриваемом случае обусловлено уменьшением количества обращений к файлу (в определенной области значений параметров процесса до $2^x – 3^x$ раз).

В современных компьютерах оперативная память позволяет хранить и обрабатывать файлы очень большого объема, на порядок уменьшилось и время доступа к внешней памяти на жестких магнитных дисках. В этих условиях пренебрежение затратами времени на выполнение операций в оперативной памяти, принятое в [1] (достаточно обосновано), оказывается неприемлемым.

Ставится задача оценки эффективности реализации алгоритма ППО на современных технических платформах, характеризующихся совершенно иными скоростными параметрами, в запоминающей среде, близкой к однородной.

2. Сущность алгоритма ППО и критерий оптимальности

Примем следующие обозначения и определения.

Основной файл ОФ – файл объемом N записей, в котором производится поиск и выборка нескольких записей по заданным ключам (идентификаторам).

Ключевой файл КФ – список K ключей искомых записей, упорядоченный синфазно с ОФ.

Активная запись – запись ОФ, ключ которой содержится в КФ.

Поле $F(i, j)$ объема $V(i, j) = j - i + 1$ – совокупность последовательных записей ОФ, заключенных в интервале порядковых номеров от i до j включительно.

ISSN 1028-9763. Математичні машини і системи, 2009, № 1

Активность $A(i, j)$ поля $F(i, j)$ – количество активных записей, входящих в поле. Если $A(i, j) = 0$, то поле $F(i, j)$ – пустое, если $A(i, j) = 1$, то поле единичное, если $A(i, j) > 1$, то поле групповое.

В смысле приведенных определений, ОФ – это групповое поле $F(1, N)$ объема N и активности K. Сущность алгоритма ППО состоит в разбиении заданного группового поля на некоторое количество производных полей (подполей) меньшего объема и последовательной проверке их активностей путем просмотров ключей граничных записей. Если в результате проверки оказывается, что проверяемое поле пустое, то над ним никакой дальнейшей обработки не производится. Если поле единичное, то активная запись ищется методом дихотомии ("бинарного" поиска [2]). Если поле групповое, то оно вновь подвергается аналогичному разбиению.

Формально рекурсивная процедура реализации алгоритма ППО включает следующую последовательность этапов, начиная с обработки поля $F(1, N)$.

1. $i := 1;\ \ j := N;\ \ A(i, j) := K;\ \ V(i, j) := N$.

2. Определение оптимального объема $V(1, j_1)$ первого подполя, вычисляемого как некоторая функция $f[A(i, j), V(i, j)]$, и номера его нижней граничной записи j_1.

3. Определение активности подполя $A(1, j_1)$. Выполняется путем последовательных сравнений ключа граничной записи j_1 с ключами КФ, начиная с первой. Если $A(1, j_1) = 0$, то выполняется п.4; иначе, если $A(1, j_1) = 1$, то выполняется п.5; иначе выполняется п.6.

4. Задача сводится к первоначальной задаче обработки поля $F(j_1 + 1, N)$ активностью K, то есть к повторению пп. 2, 3 с измененными исходными данными.

5. Активная запись ищется методом дихотомии, после чего задача сводится к обработке поля $F(j_1 + 1, N)$ активностью $K - 1$.

6. Задача сводится к обработке поля $F(1, j_1)$ активностью $(1, j_1)$, т.е. к повторению пп. 2, 3 и т.д. с исходными данными: $i := 1;\ \ j := j_1;\ \ A(i, j) := A(1, j_1);\ V(i, j) := V(1, j_1)$.

После ее решения происходит возврат на первоначальный уровень разбиения и переход к обработке поля $(j_1 + 1, N)$. При этом $i := j + 1;\ \ j := N;\ \ A(1, j) = K - A(1, j_1)$ и повторно выполняются пп. 2, 3 и т.д. Обработка подполей заканчивается, когда исчерпывается КФ и активность остающейся части ОФ становится равной нулю.

В [1] исследована более простая схема обработки, не требующая вычисления оптимального объема очередного подполя на каждом шаге. В упрощенной схеме оптимальное разбиение группового поля на m равных подполей на каждом уровне производится один раз на первом этапе и в дальнейшем изменениям не подвергается. Разница эффективностей общей и упрощенной схемы иллюстрируется следующим примером. Пусть $N = 16,\ \ K = 4$ и активными являются записи с номерами 10, 11, 14, 16. Пусть далее $f[A(i, j),\ V(i, j)] = \dfrac{V(i, j)}{A(i, j)}$. Тогда в

упрощенной схеме последовательно обрабатываются подполя (1,4), (5,8), (9,12), (13,16), а в общей схеме – подполя (1,4), (5,7), (8,9), (10,11), (12,13), (14,15), (15,16). В первом случае количество обращений к ОФ равно 12, во втором – 10. Отметим попутно, что для чисто последовательной обработки и «простого» дихотомического (бинарного) поиска этот показатель равен 16.

В [1] показано, что для упрощенной схемы при допущении о равномерном распределении активных записей и пренебрежении целочисленностью значения m минимальное количество обращений к ОФ достигается при $m_0 = K/\ln 2$. Анализ более эффективной общей схемы показывает, что минимум обращений к ОФ достигается в «узловых» точках, соответствующих целым значениям $\log_2 \frac{V(i,j)}{A(i,j)}$. Искомая узловая точка соответствует функции

$$f[A(i,j),\ V(i,j)] = \left\rfloor\ 2^{div\alpha}\right\lfloor_{б.ц.}, \qquad (1)$$

где $\alpha = \log_2 \frac{V(i,j)}{A(i,j)}$,

а $\left\rfloor\ x\ \right\lfloor_{б.ц.}$ означает ближайшее целое, большее x.

Оптимальное значение m_0 на каждом шаге разбиения равно

$$m_0 = \frac{V(i,j)}{\left\rfloor\ 2^{div\alpha}\right\lfloor_{б.ц.}}. \qquad (2)$$

В частности, если $\frac{V(i,j)}{A(i,j)} = 2^n$, $\quad (n = 1,2,...)$, то $m_o = A_{ij}$.

Для начального разбиения $V(i,j) = N,\ A(i,j) = K$, и если $\frac{N}{K} = 2^n$, то $m_0 = K.$

Соотношения (1), (2) определяют оптимальные параметры алгоритма ППО в терминах количества обращений к ОФ (т.е. без учета обработки КФ).

3. Имитационная модель и результаты моделирования

Аналитическая оценка полных затрат времени на выполнение программы ППО в однородной памяти для КФ и ОФ представляет значительные сложности. Поэтому в качестве более простого инструмента исследования выбрано имитационное моделирование процесса ППО. Целью моделирования является сравнение реальных скоростей работы программы ППО с программами дихотомического поиска и чисто последовательно совместной обработки КФ и ОФ в широком диапазоне значений коэффициента активности $a = K/N$ и разных значений коэффициента m.

При оценке результатов моделирования возникает необходимость учета двух особенностей. Первая связана с высоким быстродействием и сравнительно низкой разрешающей способностью измерения времени работы компьютера (единицы миллисекунд). Вторая особенность связана с определенной нестабильностью времени выполнения программы и, следовательно, результатов измерения при малых значениях времени.

Для учета обеих особенностей процесс моделирования для ОФ заданного объема выполняется многократно, составляя сеанс моделирования из k_1 суммируемых измерений, а сеанс повторяется k_2 раз, составляя серию сеансов, и результаты k_2 сумм усредняются.

Моделирование проведено на компьютере $P4$, язык программирования Паскаль (Delphi).

Схема имитационной модели показана на рис. 1.

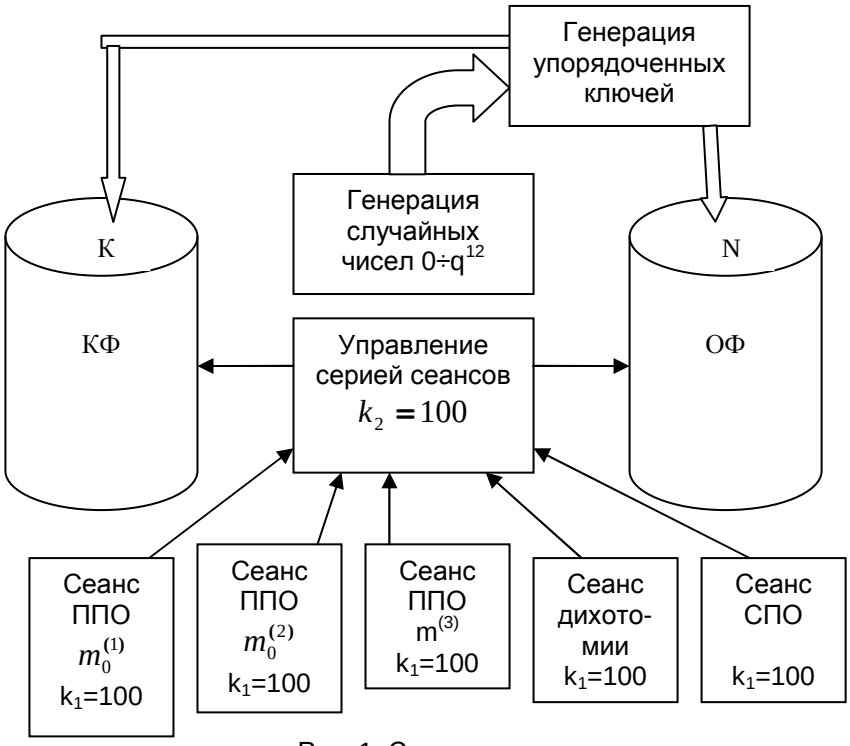

Рис. 1. Схема моделирования

Результаты моделирования отражены в табл. 1 и на рис. 2.

Таблица 1. Данные отдельной серии сеансов

a	2^{-14}	2^{-12}	2^{-10}	2^{-8}	2^{-6}	2^{-4}
τ_1	0,21333	0,22413	0,25790	0,25894	0,20103	0,14780
t_1	0:0:6	0:0:26	0:0:126	0:0:505	0:1:560	0:4:472
τ_2	0,21333	0,23305	0,26981	0,29007	0,23877	0,21641
t_2	0:0:6	0:0:28	0:0:131	0:0:564	0:1:849	0:6:542
τ_3	160,27586	38,7355	9,95072	2,59835	0,69389	0,19172
t_3	0:4: 648	0:4: 687	0: 4:846	0:5: 059	0:5:368	0:5: 791

Таблица содержит следующие данные для серии сеансов с конкретным значением $N = 5 \cdot 10^5$ и коэффициентом деления $m_0^{(1)} \approx \dfrac{A(i, j)}{\ln 2}$:

– абсолютные значения усредненной продолжительности сеансов t_1, t_2, t_3 (формат мин:сек: мсек для ППО, дихотомии и совместной последовательной обработки (СПО) соответственно при разных значениях коэффициента активности a;

– удельные значения продолжительности сеансов τ_1, τ_2, τ_3 (мсек), отнесенные к одному искомому ключу.

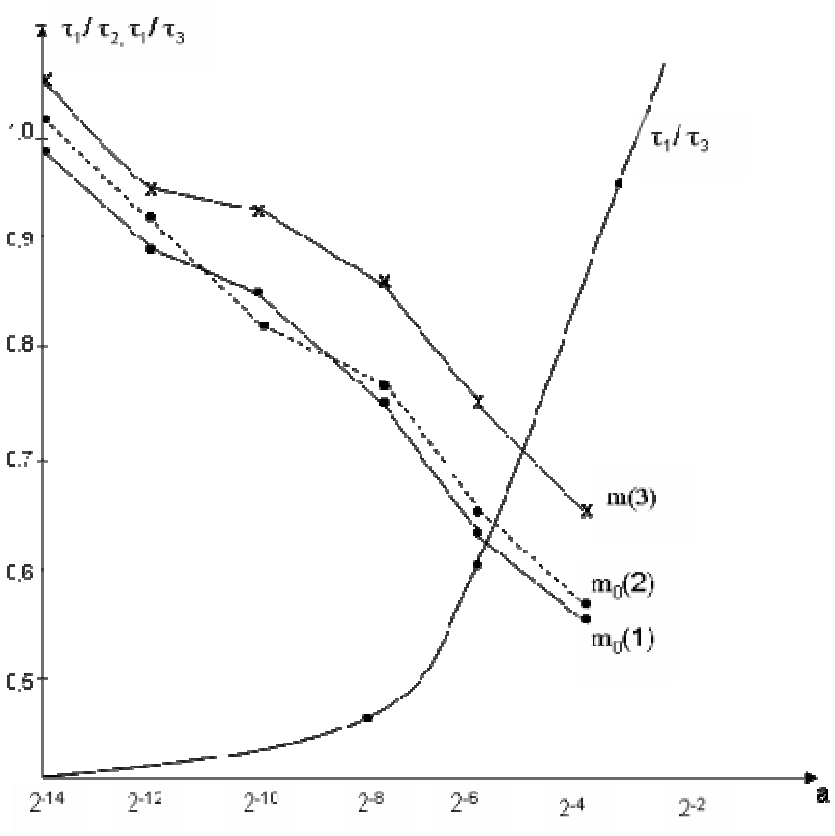

Рис. 2. Усредненные значения $\tau_1/\tau_2, \tau_1/\tau_3$

Рис. 2 представляет усредненные зависимости относительных значений τ_1/τ_2 для коэффициентов деления $m_0^{(1)} \approx \dfrac{A(i,j)}{\ln 2}$, $m_0^{(2)} \approx \dfrac{V(i,j)}{\left\lceil 2^{div\alpha} \right\rceil_{б.ц.}}$, $m^{(3)} \approx 0{,}7A(i,j)$ и τ_1/τ_3 от значений коэффициента активности для ОФ=$2\cdot10^5$, $5\cdot10^5$, $8\cdot10^5$. Значение $m^{(3)}$ выбрано произвольно для сопоставления результатов с $m_0^{(1)}$, $m_0^{(2)}$.

Проанализировав таблицу и рис. 2, можно сделать следующие выводы:

1. Алгоритм ППО обеспечивает меньшее время обработки, чем дихотомия и СПО в широком диапазоне значений a. Так, в диапазоне $a = 2^{-12} \div 2^{-6}$ отношение τ_1/τ_2 составляет величины порядка 0,9–0,65, т.е. ППО «работает» в 1,1–1,5 раз быстрее дихотомии. В таблице $a > 2^{-6}$ СПО эффективнее ППО (и, тем более, дихотомического поиска).

2. Значения $m_0^{(1)}$ и $m_0^{(2)}$ дают примерно одинаковый результат, что свидетельствует, с одной стороны, о недостаточной корректности приложения (2) к однородной памяти, а с другой стороны, о пологости гипотетической функции $f(m)$ в области минимума. Произвольно взятое значение $m^{(3)}$ дает явно худшие результаты, что косвенно подтверждает адекватность выбора оптимальных значений m.

4. Заключение

Результаты моделирования могут быть использованы для оценки и выбора метода поиска-выборки и обработки элементов последовательного файла (таблицы) по задаваемым ключам поиска в зависимости от конкретных условий: диапазона коэффициентов активности ОФ, упорядоченности ОФ и КФ, значимости возможности сокращения времени обработки.

Поскольку общий механизм сужения области поиска в B-деревьях, организующих индексы баз данных [3], подобен механизму дихотомии (m -томии), представляется возможным приложение схемы «групповой» произвольно-последовательной обработки к страницам B-деревьев для поиска K ключей-запросов.

СПИСОК ЛИТЕРАТУРЫ

1. Литвинов В.А. Алгоритм произвольно-последовательной обработки файлов // Кибернетика. – 1975. – № 5. – С. 56 – 58.
2. Кнут Д. Искусство программирования для ЭВМ. – М.: Мир, 1978. – Т. 3: Сортировка и поиск. – 845 с.
3. Кузнецов С.Д. http://www.citform.ru/programming/theory/sorting/sorting 2.shtml.

Стаття надійшла до редакції 11.12.2008