Тираж 100 экз
От печатно в московском авиационном институте
(национальном исследовательском университете)
г. Москва, А-80, ГСП-3, Волоколамское шоссе, д.4.
http://www.mai.ru//

На правах рукописи

Мьо Тант

«УПРАВЛЕНИЕ ПРОИЗВОДИТЕЛЬНОСТЬЮ ПАРАЛЛЕЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ ПРИ ОБРАБОТКЕ ЗАПРОСОВ

Специальность 05.13.11
Математическое и программное обеспечение вычислительных машин,
комплексов и компьютерных сетей

АВТОРЕФЕРАТ

диссертации на соискание учёной степени
кандидата технических наук

Москва 2012

Работа выполнена на кафедре «Вычислительные машины, системы и сети» Московского авиационного института (национального исследовательского университета, МАИ).

Научный руководитель:	доктор технических наук, профессор Брехов Олег Михайлович
Официальные оппоненты:	доктор технических наук, профессор Ильин Валерий Николаевич кандидат технических наук, доцент Сальман Леонид Абрамович
Ведущая организация:	ОАО « Научно-исследовательский институт вычислительных комплексов (НИИВК) им. М.А.Карцева»

Защита состоится «   » января 2013г. в___часов на заседании диссертационного совета Д212.125.01 при московском авиационном институте (национальном исследовательском университете) – МАИ по адресу: 125993, г. Москва, А-80, ГСП-3, Волоколамское шоссе, д.4.

С диссертацией можно ознакомиться в библиотеке московского авиационного института (национального исследовательского университета) – МАИ

Отзывы, заверенные печатью, просьба высылать по адресу: 125993, г. Москва, А-80, ГСП-3, Волоколамское шоссе, д.4, МАИ, Ученый совет МАИ

Автореферат разослан «   »_________2012г.

Ученый секретарь
диссертационного совета Д212.125.01
кандидат технических наук, доцент   А.В.Корнеенкова

В главе также представлена разработанная модель оптимизации плана выполнения запросов на GPSS/H.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

- Предложен и обоснован метод оптимизации обработки запросов для упорядоченных таблиц данных.
- Доказано утверждение об условиях определения минимального время обработки конъюнктивного запроса для упорядоченных таблиц данных.
- Проведено сравнение минимального времени обработки конъюнктивного запроса для упорядоченных и неупорядоченных таблиц данных.
- Разработан метод обеспечения оптимизации многопроцессорной обработки запросов.
- Предложен оптимальный алгоритм распределения элементарных запросов на процессоры.

Основные публикации по теме диссертации

1. Мьо Тант. Организация параллельного выполнения запросов многопроцессорной машине баз данных с иерархической архитектурой.// работы научно-практической конференции студентов и молодых ученых МАИ «Инновации в авиации и космонавтике-2010». 26-27 апреля 2010г.Москва. Тезисы докладов.-М.:Изд-во МАИ, 2010. с.52.
2. Мьо Тант. Аналитическая оценка оптимальной обработки запросов в однопроцессорной базе данных.// работы научно-практической конференции студентов и молодых ученых МАИ «Инновации в авиации и космнавтике-2012». 13-15 ноября 2012г.Москва. Тезисы докладов. – 296 с.
3. Брехов О.М., Мьо Тант. Оптимизация обработки запросов в многопроцессорной базе данных. // Журнал «Вестник Московского Авиационного Института» 2012г. №(5), с.138-265.

Таблица4

k=12				
p=(0.3,0.4,0.7,0.5,0.9,0.2,0.2,0.3,0.9.0.3,0.4,0.8)				
τ_i=[7,5,6,2,3,3,2,4,8,9,2,4]				
N процессора	Кавзиоптимальный план	Оптимальный план	Разница результата	% результата
1	2.72	2.72	0	0
2	4.285	3.66	0.625	14.58576
3	4.78	4.244	0.536	11.21339
4	5.04	4.76	0.28	5.555556
6	7.8	6.2	1.6	20.51282
12	9	9	0	0

В **четвертой главе** рассматривается разработка модуля, оценивающего время запроса. Оптимизация запросов направлена на минимизацию времени отклика для заданного запроса в данной системной среде. Каждый запрос состоит из элементарных запросов. Один запрос можно выполнить в определенной последовательности с помощью нескольких процессоров, разбив запрос на несколько более простых.

Целью разработки модуля является создание программного модуля определения оптимальной последовательности выполнения элементарных запросов при разном числе процессоров с целью минимизации времени выполнения запроса.

Входными параметрами являются: количество элементарных запросов (k), количество строк (n), вероятности выполнения элементарных запросов (p_i, i=[l, k]), времена на их обработку (τ_i, i=[1, k]),) и число процессоров (r=[1, k]). Язык программирования был выбран C++, программа написана под ОС Ubuntu.
В общем случае, формула для нахождения времени оптимального запроса для одной строки в таблице:

$$T=\tau_1+p_1*(\tau_2+p_2*(\tau_3+p_3(\tau_3+\cdots+p_{k-1}*\tau_k)\ldots),$$

Для сортировки используется соотношение:

$$\frac{\tau_1}{1-p_1}<\frac{\tau_2}{1-p_2}<\cdots<\frac{\tau_k}{1-p_k}$$

Для того, чтобы найти минимальное время из всех возможных переборов (без повторений), необходимо найти максимальное время среди каждого запроса (оно практически и будет временем запроса), затем найти минимальное среди всех возможных разделений. Также необходимо учитывать время на объединение элементарных запросов в одну конечную таблицу.

Из результатов работы программного модуля следует, что далеко не всегда использование большого числа процессоров дает лучший результат.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы. Проблеме оптимизации выполнения запросов при обращении к базе данных посвящено большое число публикаций. В качестве критерия оптимизации запросов обычно используют время выполнения запроса, при этом подразделяют время, затрачиваемое на работу с данными, находящимися в оперативной, буферной и внешней памяти. Дополнительными условиями являются объем памятей, число процессоров и др., которые часто задают в виде стоимостных ограничений.

Проблемами создания и оценки качества ООЗ занимались ряд российских и зарубежных исследователей: Григорьев Ю.А, Кузнецов С.Д, AmolDeshpande, ZaccharyIves, VijayshankarRaman, SelingerP.G, AstrahanM.M, ChamberlinD.D и др.

В данной работе в рамках базовой постановки оптимизации мы будем считать, что база данных целиком находится в основной памяти, что наиболее соответствует бортовым базам данных авиационных систем, таким как базы данных для систем управления полетом.

В простейшем случае запрос к базе данных можно рассматривать как обращение к таблице, содержащий, например, множество записей (строк), характеризующихся одинаковым множеством свойств (столбцов), с целью получения множества записей, удовлетворяющих заданному условию запроса.

Основным критерием при определении Плана реализации запроса является время выполнения запроса, которое, вообще говоря, зависит от порядка выполнения элементарных запросов, его составляющих, и от времени проверки в строке и вероятности успеха в строке. Время выполнения элементарного запроса зависит от метода доступа к столбцу таблицы. Существуют два базовых метода, когда данные в столбцах не упорядочены и когда данные в столбцах упорядочены. Известным методом увеличения производительности является использование многопроцессорных бортовых ВС.

В данной работе рассмотрена эффективность выполнения запроса в базах данных многопроцессорной бортовой вычислительной системы, что является актуальным для авиационно-космических систем.

Цель работы. Оптимизации запросов в базах данных с целью создания надстройки над СУБД, которая способна последовательно выполнять действия, необходимые для выбора наиболее эффективного плана выполнения запроса, также осуществляя минимизацию накладных расходов.

Методы исследования. Аналитическое и имитационное моделирование. Эксперименты с СУБД.

Научная новизна

- Предложен и обоснован метод оптимизации обработки запросов для упорядоченных таблиц данных.

- Проведено сравнение минимального времени обработки конъюнктивного запроса для упорядоченных и неупорядоченных таблиц данных.
- Разработан метод обеспечения оптимизации многопроцессорной обработки запросов.
- Предложен квазиоптимальный план алгоритма распределения элементарных запросов на процессоры.
- Предложен оптимальный алгоритм распределения элементарных запросов на процессоры.

Достоверность полученных в диссертационной работе результатов подтверждается:

Применяемым математическим и имитационным аппаратом. Подобием полученных результатов аналитического и имитационного моделирования. Соответствием полученных и известных результатов.

Практическая значимость. Разработан модуль формирования и оценки времени выполнения плана выполнения запроса.

Реализация результатов работы. Результаты диссертационной работы используются в учебном процессе кафедры «Вычислительные машины и системы» Московского авиационного института (государственного технического университета).

На защиту выносятся следующие положения:

- Метод обеспечения оптимизации многопроцессорной обработки запросов.
- Утверждение об условиях определения минимального время обработки конъюнктивного запроса для упорядоченных таблиц данных.
- Квазиоптимальный план распределения элементарных запросов на процессоры.
- Оптимальный план распределения элементарных запросов на процессоры.

Апробация работы. Основные положения и результаты диссертационного исследования докладывались автором и обсуждались: на всероссийской конференции молодых ученых и студентов «Информационные технологии в авиационной и космической технике», Москва, 2010г. и трех Международных конференциях «Авиация и космонавтика», Москва, 8–10 ноября 2011 года, 17–20 апреля и 13-15 ноября 2012 года.

Публикации. Основные материалы диссертационной работы опубликованы в 3 печатных работах.

Структура и объем диссертации. Диссертация состоит из введения, четырёх глав, заключения, библиографического списка из 142 наименований и одно приложение. Работа изложена на 127 страницах, содержит 23 таблиц и 34 рисунка.

6	1,12	3.6	7.8
	2,11	3.2	
	3,10	3.8	
	4,9	5	
	5,8	6.7	
	6,7	7.8	
12	1 до 12		9

Оптимальный алгоритм распределения ЭЗ на процессоры

Оптимальный алгоритм распределения ЭЗ при выше приведенных данных дает результаты, приведенные в таблице 3.

Таблица3

k=12			
p=(0.3,0.4,0.7,0.5,0.9,0.2,0.2,0.3,0.9.0.3,0.4,0.8)			
τ_i=[7,5,6,2,3,3,2,4,8,9,2,4]			
N процессора	Порядок обработки ЭЗ		
1	1,2,3,4,5,6,7,8,9,10,11,12	-	2.72
2	11,6,4,2,12,5	3.662	3.66
	7,8,1,10,3,9	3.444	
3	6,8,2,3	4.244	4.24 4
	11,4,12,5	4.08	
	7,1,10,9	4.084	
4	4,8,5	4.45	4.76
	2,7,10	4.76	
	6,1,3	4.64	
	11,2,12	4.28	
6	2,5	6.2	6.2
	8,12	5.2	
	4,9	6	
	6,3	4.2	
	11,10	5.6	
	7,1	3.4	
12	1 до 12		9

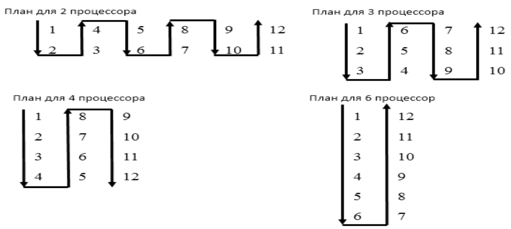
В таблице 4 приведено сравнение для неупорядоченной таблицы при квазиоптимальном и оптимальном планах, т.е. эффективность оптимального плана составляет от 5 до 20 %.

где T_{rj}, j =1,...,r, - время выполнения j-го подмножества элементарных запросов на j-м из r процессоров.

В зависимости от алгоритма распределения элементарных запросов ЭЗ на процессоры время завершения T_r первого этапа может быть разным.

В качестве квазиоптимального алгоритма распределения элементарных запросов на процессоры для случая, когда число процессоров r отвечает условию [k/r]r = k, предлагается алгоритм чередования, при котором i–ый (i =1,...,r) процессор получает элементарные запросы с номерами i, 2r +1 - i, 2r + i, 4r +1 - i, 4r + i, 6r +1 - i, 6r + i,….

Например, для k = 12 распределение имеет вид:



Время выполнения запроса при значениях параметров τ_i и p_i: приведено в таблице 2.

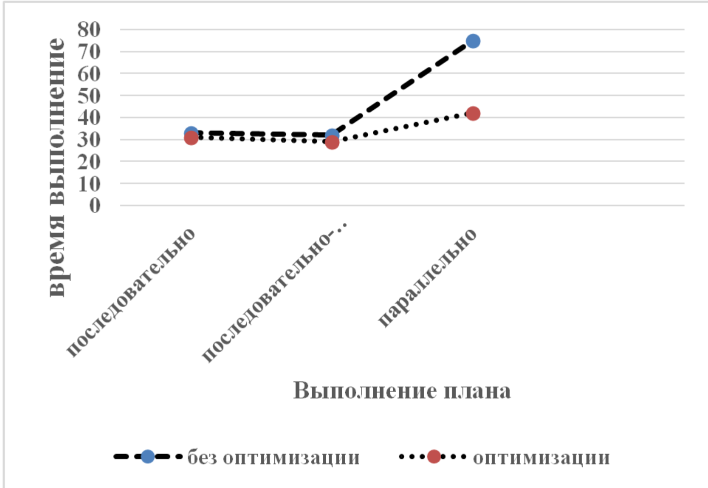
Таблица2

k=12			
p=(0.3,0.4,0.7,0.5,0.9,0.2,0.2,0.3,0.9.0.3,0.4,0.8)			
τ=[7,5,6,2,3,3,2,4,8,9,2,4]			
N процессора	Порядок обработки ЭЗ		T_r
1	1,2,3,4,5,6,7,8,9,10,11,12	-	2.73
2	1,4,5,8,9,12	3.174	4.285
	2,3,6,7,10,11	4.285	
3	1,6,7,12	3.75	4.78
	2,5,8,11	4.78	
	3,4,9,10	4.28	5.04
4	1,8,9	4.16	
	2,7,10	5.04	
	3,6,11	4.24	
	4,5,12	4.48	

СОДЕРЖАНИЕ РАБОТЫ

Во **введении** обоснована актуальность выполненного исследования сформулированы цель, задач диссертационной работы, научная новизна, практическая ценность диссертационной работы по главам.

В **первой главе** сформулирована постановка задачи и описание подхода к ее решению. Приведено описание процесса обработки запросов в СУБД, проанализированы компоненты ядра СУБД, участвующие в обработке запросов пользователей. Проведен анализ методов оптимальной обработки запросов в многопроцессорных базах данных. Выделены аспекты, определяющие процесс оптимизации запросов, приведена общая классификация подходов к оптимизации запросов, реализуемых в современных СУБД. Проведён анализ основных подходов к оптимизации запросов с точки зрения реализуемых стратегий поиска оптимального плана выполнения запроса. Проведён анализ существующих методов оценки стоимости операции соединения таблиц базы данных. На основе экспериментов показано, что в зависимости от порядка реализации элементарных запросов минимальное время реализации плана достигается для параллельно-последовательного плана.



В результате сформулирована основная задача исследования - оптимизация плана реализации запроса в многопроцессорных базах данных с учетом использования последовательного, параллельного и параллельно-последовательного методов.

Во **второй главе** разработан метод оптимизации однопроцессорной обработки запросов.

В простейшем случае запрос к базе данных можно рассматривать как обращение к таблице, содержащий, например, множество записей (строк), характеризующихся одинаковым множеством свойств (столбцов), с целью получения множества записей, удовлетворяющих заданному условию запроса. Будем называть часть запроса, относимую к i– му свойству записи (i – му столбцу), i – ым элементарным запросом ЭЗ$_i$.

При выполнении запроса для каждой строки необходимо выполнить проверку условия, связанного с ЭЗ$_i$, требуемое для этого время обозначим через τ_i вероятность успеха выполнения условия через p_i. Вариация значений времени τ_i зависит от формулировки условия ЭЗ$_i$ (точечное, интервальное или более сложное условие), а так же от технических и/или программных особенностей реализации (кэширование, диски и др.). Вариация значений вероятности выполнения условия p_i определяется содержимым базы данных.

Основным критерием при определении Плана реализации запроса является время выполнения запроса, которое, вообще говоря, зависит от порядка выполнения элементарных запросов, его составляющих, и указанных параметров элементарных запросов ЭЗ$_i$ - времени проверки в строке τ_i и вероятности успеха в строке pi.

Время выполнения элементарного запроса зависит от метода доступа к столбцу таблицы. Мы рассмотрим два базовых метода, когда данные в столбцах неупорядочены и когда данные в столбцах упорядочены построчно. В первом случае время выполнения элементарного запроса ЭЗ$_i$ определяется временем проверки всех r строк i-го столбца таблицы и равно r τ_i. Во втором случае, например, при использовании индексной организации обращений к данным, определяется временем проверки r p_iстрок и равно r $p_i\tau_i$.

Время выполнения запроса определяется Теоремами 1 и 2.

Теорема 1. Минимальное время $T_{\min \text{уп}}$ обработки запроса для упорядоченных столбцов таблицы, состоящего из k конъюнкций элементарных запросов (ЭЗ), достигается при выполнении элементарных запросов в порядке 1,2,....,k, который обусловлен выполнением условия:

$$\frac{p_i\,\tau_i}{1-p_i} < \frac{p_{i+1}\,\tau_{i+1}}{1-p_{i+1}}, \qquad i = \overline{1, k-1,}$$

при этом минимальное время

$$T_{\min \text{уп}} = n\sum_{i=1}^{k} \tau_i \prod_{j=1}^{i} p_j,$$

где

n- число строк таблицы базы данных,

k - число столбцов таблицы,

τ_i- время обработки i-го ЭЗ для одной строки таблицы,

p_i - вероятность успеха (данные отвечают заданному условию) при обработке i-го ЭЗ для одной строки таблицы.

Доказательство проведем по индукции.

1. Пусть k =2.

$$T_{3\text{нг}} = \max_{j=1,3} T_{3j\text{нг}} = \mathrm{T}_{33\text{нг}} = \mathrm{n(a)}^2\,\frac{1-(\mathrm{p(a)}^3)^4}{1-\mathrm{p(a)}^3}.$$

Для упорядоченной таблицы оптимальное время выполнения запроса в соответствии с теоремой 1 при изменении параметра времени τ_i по закону геометрической прогрессии для r процессоров равно:

$$\mathrm{Trjуг} = np(a)^{j-1}\sum_{i\in 0,\frac{k}{r}-1}(p(a)^r)^i = np\,\frac{1-(p(a)^r)^{\frac{k}{r}}}{1-p(a)^r}.\ \mathrm{j}=1,...,\mathrm{r}.$$

Для k=12 и r =3 имеем:

$$T_{3\text{уг}} = \max_{j=1,3} T_{3j\text{уг}} = \mathrm{T}_{33\text{уг}} = \mathrm{np(a)}^2\,\frac{1-(\mathrm{p(a)}^3)^4}{1-\mathrm{p(a)}^3}.$$

Для неупорядоченной таблицы оптимальное время выполнения запроса в соответствии с теоремой 2 при изменении параметра времени τ_i по закону арифметической прогрессии для r процессоров на первом этапе равно:

$$T_{rj\text{на}} = \mathrm{n}\Big(\frac{1-\mathrm{p}^{k/r}}{1-\mathrm{p}}\ (1+(\mathrm{j}-1)\ \Delta) + \mathrm{rp}\Delta\frac{1-(\frac{\mathrm{k}}{\mathrm{r}})\mathrm{p}^{k/r-1}+(k/r-1)\mathrm{p}^{k/r}}{(1-\mathrm{p})^2}\Big)$$

$$T_{r\text{на}} = \max_{j=1,r} T_{rj\text{на}} = \mathrm{T}_{\mathrm{rr}\text{на}}.$$

Для k=12 и r =3 имеем:

$$T_{3\text{на}} = \max_{j=1,3} T_{3j\text{на}} = T_{33\text{на}} = \mathrm{n}\ \Big(\frac{1-\mathrm{p}^4}{1-\mathrm{p}}\ (1+2\Delta\ ) + 3\mathrm{p}\Delta\frac{1-4\mathrm{p}^3+3\mathrm{p}^4}{(1-\mathrm{p})^2}\Big).$$

Для упорядоченной таблицы оптимальное время выполнения запроса в соответствии с теоремой 1 при изменении параметра времени τ_i по закону арифметической прогрессии для r процессоров на первом этапе равно:

$$T_{rj\text{уа}} = \mathrm{np}\Big(\frac{1-\mathrm{p}^{k/r}}{1-\mathrm{p}}\ (1+(\mathrm{j}-1)\ \Delta) + \mathrm{rp}\Delta\frac{1-(\frac{\mathrm{k}}{\mathrm{r}})\mathrm{p}^{k/r-1}+(k/r-1)\mathrm{p}^{k/r}}{(1-\mathrm{p})^2}\Big),$$

$$T_{r\text{уа}} = \max_{j=1,r} T_{rj\text{уа}} = \mathrm{T}_{\mathrm{rr}\text{уа}}.$$

Для k=12 и r =3 имеем:

$$T_{3\text{уа}} = \max_{j=1,3} T_{3j\text{уа}}$$

$$T_{3\text{уа}} = \max_{j=1,3} T_{3j\text{уа}} = T_{33\text{уа}} = \mathrm{np}\Big(\frac{1-\mathrm{p}^4}{1-\mathrm{p}}\ (1+2\Delta\ )\ +\ 3\mathrm{p}\Delta\,\frac{1-4\mathrm{p}^3+3\mathrm{p}^4}{(1-\mathrm{p})^2}\Big).$$

Полученные результаты демонстрируют одну и ту же тенденцию – при значении параметра вероятности $p_i >= 0.5$ минимальное время выполнения запроса достигается не всегда при максимальном значении числа процессоров.

Квазиоптимальный план алгоритма распределения ЭЗ на процессоры

При параллельном использования r процессоров на первом этапе будут получены r таблиц, которые на втором этапе посредством объединения по номеру (имени) образуют результирующую таблицу.

Существует задача формирования подмножеств элементарных запросов для достижения минимального значения времени завершения T_r первого этапа, которое равно:

$T_r = \max_j T_{rj}$, j=1,…,r.

объединения строк заканчивается, при этом строка исключается из объединенной таблицы.

В этих условиях верхняя оценка среднего числа обращений к таблицам при формировании одной строки заключительной таблицы равна:

$\mathrm{M} = 1(t_{\text{п}} + t_{\text{ф}}) + 1(1-\mathrm{q})t_{\text{п}} + \mathrm{q}(1-\mathrm{q}) * \left((t_{\text{п}} + t_{\text{ф}}) + t_{\text{п}}\right) + q^2(1-\mathrm{q}) * \left(2(t_{\text{п}} + t_{\text{ф}}) + t_{\text{п}}\right) + \dots + q^{r-2}(1-\mathrm{q})((\mathrm{r}-2)\ (t_{\text{п}} + t_{\text{ф}}) + t_{\text{п}})) + q^{r-1}((\mathrm{r}-1) * (t_{\text{п}} + t_{\text{ф}})) = \frac{1-q^r}{1-q} * (t_{\text{п}} + t_{\text{ф}}) + (1-q^{r-1})t_{\text{п}}.$

Тогда среднее время объединения всех строк r таблиц равно:

То= М np^r.

Поэтому общее время выполнения запроса в r - процессорной ВС равно:

T= Tr+ To

Влияние числа процессоров на время выполнения запроса

Для оценки влияния изменения числа процессоров на изменение времени выполнения запроса рассмотрим два закона задания функций изменения параметров времениτ_i (геометрической прогрессии и арифметической прогрессии) при двух базовых методах доступа к столбцам таблицы, когда данные в столбцах не упорядочены и упорядочены.

Пусть запрос образуют конъюнкции k элементарных запросов:

1. С изменением параметра времени по закону геометрической прогрессии$\tau_i = a^{i-1}$,и с постоянным значением параметра вероятности$p_i = p, i \in 1, k$.
2. С изменением параметра времени по закону арифметической прогрессииτ_i = 1+(i-1)Δ, и с постоянным значением параметра вероятности$p_i = p, i \in 1, k$.

В главе 2 показано, что параметр вероятности p_i существенновлияет на время выполнения запроса при значениях более 0.5, поэтому влияние изменения числа процессоров на изменение времени выполнения запроса мы рассмотрим при 0.5< p_i < 0.9. При этом, исходя из практических условий вариации значений времени τ_i, диапазон изменения времени τ_i пусть находится в интервале от1 до 4, что, в частности, при k =6 приводит к a = 1.2 или 1.3, т.к. 1.25 = 2.488 или 1.35 = 3.713, а Δ = 0.5, и при k =12 приводит к a = 1.1, т.к. 1.111= 2.853, а Δ = 0.2.

Для неупорядоченной таблицы оптимальное (Минимальное) время выполнения запроса в соответствии с теоремой 2 при изменении параметра времени τ_i по закону геометрической прогрессии для r процессоров равно:

Trjнг= $n(a)^{j-1} \sum_{i \in 0, \frac{k}{r}-1} (p(a)^r)^i = n \frac{1-(p(a)^r)^{\frac{k}{r}}}{1-p(a)^r}$. j =1,...,r.

Trнг = $\max_{j=1,3} T_{3j}$=$T_{rr\text{нг}}$.

Для k=12 и r =3 имеем:

При обработке элементарных запросов в порядке 1, 2 получаем, что при обработке 1-го ЭЗ потратим время τ_1 (np_1), при этом остается np_1строк, отвечающих требованию 1-го ЭЗ. При обработке 2-го ЭЗ потратим время $\tau_2(np_1p_2)$. Суммарное время обработки запроса равно $\mathrm{T}_1 = n(p_1\tau_1 + p_1p_2\tau_2)$. Аналогично, суммарное время при обработке элементарных запросов в порядке 2 и 1 равно $\mathrm{T}_2 = n(p_2\tau_2 + p_1p_2\tau_1)$.

Очевидно, что $\mathrm{T}_1 < \mathrm{T}_2$ при обработке запросов в порядке 1,2, когда выполняется условие $\frac{p_1\tau_1}{1-p_1} < \frac{p_2\tau_2}{1-p_2}$, при этом имеем минимальное время $T_{min} = n(p_1\tau_1 + p_1p_2\tau_2)$.

2. Пусть к=3.

Возможные 6 последовательностей обработки запроса, состоящего из 3-х элементарных запросов, определим посредством строк таблицы, в которых i-й последовательности соответствует порядок обработки элементарных запросов и время выполнения запроса, указанный в таблице 1.

Таблица 1

Номер последовательности	Последовательность	Время выполнения
1	1 2 3	$\mathrm{T}_1 = n(p_1\tau_1 + p_1p_2\tau_2 + p_1p_2p_3\tau_3)$
2	1 3 2	$\mathrm{T}_2 = n(p_1\tau_1 + p_1p_3\tau_3 + p_1p_2p_3\tau_2)$
3	2 1 3	$\mathrm{T}_3 = n(p_2\tau_2 + p_1p_2\tau_1 + p_1p_2p_3\tau_3)$
4	2 3 1	$\mathrm{T}_4 = n(p_2\tau_2 + p_2p_3\tau_3 + p_1p_2p_3\tau_1)$
5	3 1 2	$\mathrm{T}_5 = n(p_3\tau_3 + p_1p_3\tau_1 + p_1p_2p_3\tau_2)$
6	3 2 1	$\mathrm{T}_6 = n(p_3\tau_3 + p_2p_3\tau_2 + p_1p_2p_3\tau_1)$

Из сравнения последовательностей 1 и 2 следует, что время выполнения T_1меньше времени выполнения T_2 тогда и только тогда, когда выполняется условие

$\frac{p_2\tau_2}{1-p_2} < \frac{p_3\tau_3}{1-p_3}$.

Аналогично,

$\mathrm{T}_3 < \mathrm{T}_4$ тогда и только тогда, когда $\frac{p_1\tau_1}{1-p_1} < \frac{p_3\tau_3}{1-p_3}$,

$\mathrm{T}_5 < \mathrm{T}_6$ тогда и только тогда, когда $\frac{p_1\tau_1}{1-p_1} < \frac{p_2\tau_2}{1-p_2}$.

Кроме этого, для последовательностей 1,3 и последовательностей 2,5 имеем:

$\mathrm{T}_1 < \mathrm{T}_3$ тогда и только тогда, когда $\frac{p_1\tau_1}{1-p_1} < \frac{p_2\tau_2}{1-p_2}$.

$\mathrm{T}_2 < \mathrm{T}_5$ тогда и только тогда, когда $\frac{p_1\tau_1}{1-p_1} < \frac{p_3\tau_3}{1-p_3}$.

Следовательно, минимальное время $T_1 = \min(T_i) = n(p_1\tau_1 + p_1p_2\tau_2 + p_1p_2p_3\tau_3)$ тогда и только тогда, когда $\frac{p_1\tau_1}{1-p_1} < \frac{p_2\tau_2}{1-p_2} < \frac{p_3\tau_3}{1-p_3}$.

3. Пусть теорема 1 справедлива для случая, когда запрос состоит из (k-1)-го элементарных запросов, для которых заданы параметры: $n, \tau_i = \overline{1,k}, p_i = \overline{1,k}$.

4. Докажем, что теорема справедлива для запроса из k элементарных запросов.

Зададим порядок обработки конъюнкции элементарных запросов посредством числовых последовательностей, которые пронумеруем последовательно от 1 до k! включительно.

Обозначим время обработки запроса с номером последовательности l через T_l.

Среди (k-1)! последовательностей, начинающихся с элементарного запроса 1, минимальное время обработки запроса определяется последовательностью с номером 1, для которой.

$T_1 = \mathrm{n}p_1\tau_1 + p_1 n \left(\sum_{i=2}^{k} \tau_i \prod_{j=2}^{i} p_j\right)$,

т.к. для элементарных запросов i, i=2, k, выполняется предположение пункта 3 (n=k-1) при выполнении условия

$\frac{p_i\tau_i}{1-p_i} < \frac{p_{i+1}\tau_{i+1}}{1-p_{i+1}}$, i=$\overline{2,(k-1)}$,

т.е. минимальное время

$T_1 = \min(T_l), l = \overline{1,(k-1)!}$.

Среди (k-1)! последовательностей, начинающихся со ЭЗ$_2$, минимальное время обработки запроса достигается для последовательности с номером (1+(к-1)!), т.е. минимальное время

$T_{1+(\mathrm{к}-1)!} = \min_{l \in (1+(\mathrm{k}-1)!, 2(\mathrm{k}-1)!)} T_l$,

при выполнении условия

$\frac{p_1\tau_1}{1-p_1} < \frac{p_3\tau_3}{1-p_3} < \frac{p_4\tau_4}{1-p_4} < \cdots < \frac{p_k\tau_k}{1-p_k}$.

Аналогично, среди (к-1)!-ой последовательностей, начинающихся с i-ого ЭЗ, имеем минимальное время:

$T_{1+(i-1)(k-1)!} = \min_{l \in (1+(\mathrm{i}-1)\mathrm{k}-1!, \mathrm{i}(\mathrm{k}-1)!)} T_l$,

при выполнении условия

$\frac{p_1\tau_1}{1-p_1} < \cdots < \frac{p_{i-1}\tau_{i-1}}{1-p_{i-1}} < \frac{p_{i+1}\tau_{i+1}}{1-p_{i+1}} < \cdots < \frac{p_k\tau_k}{1-p_k}$,

где i =$\overline{2, k-1}$.

Наконец, для последовательностей, начинающихся с ЭЗ$_k$, имеем минимальное время:

$T_{1+(k-1)(k-1)!} = \min_{l \in (1+(\mathrm{k}-1)\mathrm{k}-1!, \mathrm{k}!)} T_l$,

при выполнении условия

$\frac{p_i\tau_i}{1-p_i} < \frac{p_{i+1}\tau_{i+1}}{1-p_{i+1}}, i = \overline{1, k-2}$.

В общем случае запрос Q образует множество d дизъюнкций конъюнкций элементарных запросов:

Q = CEQ1 v… v CEQiv…v CEQd,

где

конъюнкция

CEQi= EQ1 ^… ^ EQj^…^ EQki,

EQj– элементарный запрос

Время выполнения запроса Q равно сумме времени выполнения всех конъюнкций элементарных запросов, в соответствии с теоремами 1 и/или 2, плюс время объединения всех d таблиц, полученных на этапе реализации d конъюнкций.

В **третьей главе** рассматривается время выполнения запроса в многопроцессорной ВС.

Будем считать, что нумерация образующих конъюнкцию элементарных запросов отвечает условиям теоремы 1 (или теоремы 2).

С целью уменьшения общего времени выполнения запроса широко используют многопроцессорную обработку. При k элементарных запросах может использоваться от 1 до k процессоров, при этом процессоры параллельно должны выполнять подмножество множество элементарных запросов в порядке, обусловленным теоремой 1 (или теоремой 2).

В случае использования r процессоров на первом этапе будут получены r таблиц, которые на втором этапе посредством объединения по номеру (имени) образуют результирующую таблицу. Поэтому время выполнения запроса определяется временем выполнения первого и второго этапов, которые, вообще говоря, могут быть совмещены для уменьшения суммарного времени, однако здесь будем считать, что совмещение этапов отсутствует.

ОбозначимT_{rj}, j =1,...,r, - время выполнения j-го подмножества элементарных запросов на j-м из r процессоров, при этом формируется r таблиц. Пусть также Tr – время, требуемое для завершения параллельной работы r процессоров, т.е. для завершения первого этапа, тогда$T_r = \max_j T_{rj}, j = 1, \dots, r$. Существует задача формирования подмножеств элементарных запросов для достижения минимального значения времени завершения первого этапа.

На втором этапе посредством объединения по номеру (имени) образуют результирующую таблицу. При объединении этих таблиц, по $s = np^r$строк в каждой таблице, где вероятность $q = p^{k/r}$для каждой строки будем считать постоянной, что и задает верхнюю границу.

Требуется выполнить два шага. На первом шаге необходимо определить за время поиска $t_\text{п}$ имеется ли одноименная строка в текущей таблице. Если строка с заданным именем присутствует (вероятность q), то после выполнения второго шага – формирование строки за время $t_\text{ф}$, процесс объединения строк продолжается. Иначе (с вероятностью (1- q)) процесс формирования закона задания функций изменения параметров времени τ_i (геометрической прогрессии и арифметической прогрессии) при двух базовых методах доступа к столбцу таблицы, когда данные в столбцах неупорядочены и упорядочены.

Для упорядоченной и неупорядоченной таблиц минимальное время выполнения запроса в соответствии с теоремами 1 и 2 при изменении параметра времени τ_i по закону геометрической прогрессии равно, соответственно:

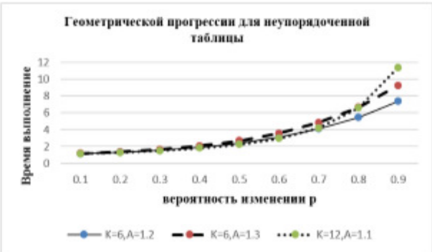
$$T_{\min\text{ уп}} = np\frac{1-(ap)^k}{1-ap}, \qquad T_{\min\text{ ну}} = n\frac{1-(ap)^k}{ap}$$

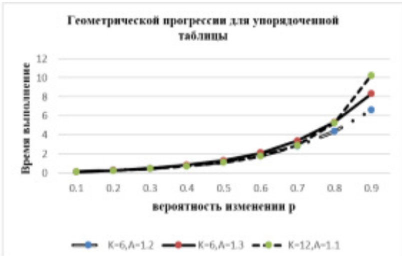
Для упорядоченной и неупорядоченной таблиц минимальное время выполнения запроса в соответствии с теоремами 1 и 2 при изменении параметра времени τ_i по закону арифметической прогрессии равно, соответственно:

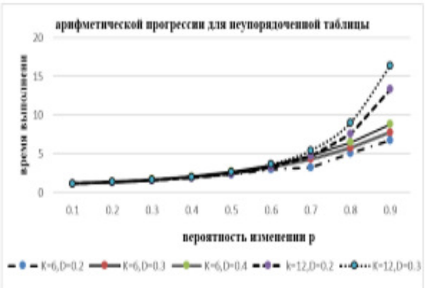
$$T_{1\text{ану}} = n\left(\frac{1-p^k}{1-p} + p\Delta\frac{1-kp^{k-1}+(k-1)p^k}{(1-p)^2}\right),$$

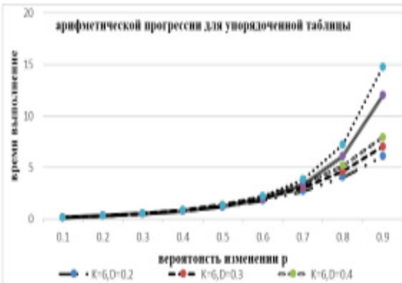
$$T_{1ауп} = np\left(\frac{1-p^k}{1-p} + p\Delta\frac{1-kp^{k-1}+(k-1)p^k}{(1-p)^2}\right)$$

Представленные выше результаты для параметров времени a=1.2 или 1.3, Δ=0.5при k=6, и a= 1.1,Δ=0.2 при k=12, демонстрируют одну и ту же тенденцию – при изменении параметра вероятностир$_i$ до значения 0.5 время выполнения запроса возрастает незначительно. Параметр вероятности p$_i$ существенно влияет на время выполнения запроса при значениях более 0.5.









Таким образом, минимальное время обработки запросов последовательностей, начинающихся с ЭЗ$_i$, достигается при выполнении соответствующих неравенств для последовательностей элементарных запросов с номерами (1+(i-1)(k-1)!, i=1, k, т.е. эти k последовательностей являются доминантными для (k-1)! своих (с первыми номерами ЭЗ$_i$) последовательностей.

В свою очередь, соответствующие (k-1) последовательности с номерами (1+(i-2)(k-2)!, i=2, k, из группы последовательностей, начинающихся с ЭЗ$_1$, являются доминантными для выше указанных последовательностей с номерами (1+(i-1)(k-1)!, i=2, k, при выполнении соответствующих неравенств.

В самом деле, время выполнения последовательностей с номерами 1 и (1+(k-1)! равны соответственно:

$$T_1 = n(p_1\tau_1 + p_1p_2\tau_2 + \sum_{i=3}^{k}\tau_i\prod_{j=1}^{i}p_j),$$

$$T_{1+(k-1)!} = n(p_2\tau_2 + p_1p_2\tau_1 + \sum_{i=3}^{k}\tau_i\prod_{j=1}^{i}p_j),$$

Из сравнения этих выражений следует, что T1< T1+(k-1)! выполняется тогда и только тогда, когда выполняется условие

$\frac{p_1\tau_1}{1-p_1} < \frac{p_2\tau_2}{1-p_2}$,

и для последовательностей 1+i(k-2)! и (1+(i+1)(k-1)! , i=$\overline{0,k-2}$,

T1+i(k-2)! <T1+(i+1)(k-1)!

выполняется тогда и только тогда, когда выполняется условие

$\frac{p_1\tau_1}{1-p_1} < \frac{p_j\tau_j}{1-p_j}, j = \overline{2,k}.$

Следовательно, минимальное время

$T_1 = \min(T_l)\,, i = \overline{1,k!}$

выполняется тогда и только тогда, когда выполняется условие

$\frac{p_i\tau_i}{1-p_i} < \frac{p_{i+1}\tau_{i+1}}{1-p_{i+1}}\,, i = \overline{1,k-1},$

при этом минимальное время имеет вид:

$T_{\min\text{ ну}} = T_1 = n\sum_{i=1}^{k}\tau_i\prod_{j=1}^{i}p_j,$

что и т. д.

Время выполнения запроса для неупорядоченных столбцов таблицы.

Теорема2. Минимальное время $T_{\min\text{ ну}}$ обработки запроса для неупорядоченных столбцов таблицы, состоящего из k конъюнкций элементарных запросов (ЭЗ), достигается при выполнении элементарных запросов в порядке 1,2,....,k,:

$$\frac{\tau_i}{1-p_i} < \frac{\tau_{i+1}}{1-p_{i+1}}, \qquad i = \overline{1,k-1},$$

при этом

$T_{\min\text{ ну}} = n(\tau_1 + \sum_{i=2}^{k}\tau_i\prod_{j=1}^{i-1}p_j),$

где

n- число строк таблицы базы данных,

k - число столбцов таблицы,

τ_i- время обработки i-го ЭЗ для одной строки таблицы,
p_i- вероятность успеха (данные отвечают заданному условию) при обработке i-го ЭЗ для одной строки таблицы.
Доказательство выполняется по индукции аналогично доказательству теоремы 1.

Сравнение минимального времени обработки запроса для упорядоченных и неупорядоченных столбцов таблицы

Сравнение минимального времени обработки запроса для упорядоченных и неупорядоченных столбцов таблицы определяются Следствиями 1 и 2.

Следствие 1. Условия порядка обработки: $\frac{\tau_i}{1-p_i} < \frac{\tau_{i+1}}{1-p_{i+1}}$ и $\frac{p_i \tau_i}{1-p_i} < \frac{p_{i+1} \tau_{i+1}}{1-p_{i+1}}$, $i = \overline{1, k-1}$, в общем случае могут задать различный порядок выполнения элементарных запросов для неупорядоченных и упорядоченных столбцов таблицы.

Доказательство следует на основе двух частных примеров:

1. Для значений $k = 2$, p_1=0.8, $\tau_1 = 1$ и p_2=0.6, $\tau_2 = 2.2$

получаем различный оптимальный порядок выполнения.
Для неупорядоченной таблицы оптимальный порядок выполнения элементарных запросов имеет вид 1,2, $\text{т. к. } \frac{\tau_1}{1-p_1} = 5, \text{что меньше, чем } \frac{\tau_2}{1-p_2} = 5.5$
Для упорядоченной таблицы оптимальный порядок выполнения элементарных запросов имеет вид $2{,}1, \text{т. к. } \frac{p_1 \tau_1}{1-p_1} = 4, \text{что больше, чем } \frac{p_2 \tau_2}{1-p_2} = 3.3.$

2. Для значений $k = 2$, p_1=0.8, $\tau_1 = 1.5$ и p_2=0.6, $\tau_2 = 2.2$

получаем одинаковый оптимальный порядок выполнения.
Для неупорядоченной таблицы оптимальный порядок выполнения элементарных запросов имеет вид 2,1, т.к. $\frac{\tau_1}{1-p_i} = 7.5, \text{что больше, чем } \frac{\tau_2}{1-p_2} = 5.5$, аналогичный порядок сохраняется для упорядоченной таблицы, т.к. $\frac{p_1 \tau_1}{1-p_1} = 6, \text{что больше, чем } \frac{p_2 \tau_2}{1-p_2} = 3.3.$

Следствие 2. Минимальное время $T_{\min \text{уп}}$ обработки запроса, состоящего из k конъюнкций элементарных запросов, достигается при использовании таблиц с упорядоченными столбцами при выполнении элементарных запросов в порядке 1,2,....,k, в соответствии с условием:

$$\frac{p_i \tau_i}{1-p_i} < \frac{p_{i+1} \tau_{i+1}}{1-p_{i+1}}, \qquad i = \overline{1, k-1},$$

при этом

$$T_{\min \text{уп}} = n \sum_{i=1}^{k} \tau_i \prod_{j=1}^{i} p_j.$$

Доказательство следует при сохранении порядка выполнения элементарных запросов для неупорядоченных и упорядоченных столбцов таблицы непосредственно из сравнения выражений для минимального времени обработки запроса. При не сохранении порядка выполнения элементарных запросов для неупорядоченных и упорядоченных столбцов таблицы требуется

дополнительное доказательство, которое, например, для $k = 2$имеет следующий вид.
Пусть
$\frac{\tau_1}{1-p_1} < \frac{\tau_2}{1-p_2}, \ \frac{p_2 \tau_2}{1-p_2} < \frac{p_1 \tau_1}{1-p_1}.$
Тогда
$\frac{p_1 \tau_1}{1-p_1} > p_2 \frac{\tau_2}{1-p_2} > p_2 \frac{\tau_1}{1-p_1}.$
Откуда $p_1 > p_2$.
Отсюда следует справедливость неравенства $n(p_1\tau_2 + p_1p_2\tau_1) < n(\tau_1 + p_1\tau_2)$, т.к. $0 < \tau_1(1 - p_1p_2) + \tau_2(p_1 - p_2)$, т. е. справедливость Следствия 2 .

Эффективность метода оптимизации

Эффективность предложенного метода оптимизации продемонстрируем на простом примере. Пусть запрос образуют конъюнкции k элементарных запросов с параметрами $\tau_i = a^{i-1}$, $p_i = p, i \in 1, k$.

Минимальное время выполнения запроса для неупорядоченной таблицы в соответствии с теоремой 2 равно

$$T_{\min} = n \cdot \sum_{i \in 0,k-1} (ap)^i = n \cdot \frac{1-(ap)^k}{1-ap}.$$

При выполнении обработки элементарных запросов в обратном порядке, находим:

$$T_{\text{обр}} = n\left(a^{k-1} + p \cdot a^{k-2} + \dots + p^{k-2} \cdot a + p^{k-1} \cdot 1\right) =$$
$$= na^{k-1} \sum_{i \in 0,k-1} (p/a)^i = na^{k-1} \frac{1-(p/a)^k}{1-p/a}.$$

Поэтому для упорядоченной и неупорядоченной таблиц при изменении параметра времен $\tau_i = a^{i-1}$и постоянного значения параметра вероятности $p_i = p, i \in 1, k$, и эффективность оптимального порядка обработки элементарных запросов лучше в C раз по отношению к обратному порядку обработки элементарных запросов:

$$C = \frac{T_{\text{обр}}}{T_{\min}} = \frac{a^{k-1} \frac{1-(p/a)^k}{1-p/a}}{\frac{1-(ap)^k}{1-ap}}.$$

Тогда, при k=12, a=1.1 и p=0.7 получаем:

$$T_{\min} == n \cdot \frac{1-(0.77)^{12}}{1-0.77} = 4.16n, \qquad T_{\text{обр}} = n \cdot 1.1^{11} \cdot \frac{1-(\frac{7}{11})^{12}}{1-\frac{7}{11}} = 7{,}81n, C = 1{,}88.$$

т.е. Эффективность оптимального порядка обработки элементарных запросов по отношению к обратному порядку обработки элементарных запросов здесь лучше в 1,88 раз.

Для оценки влияния изменения значений параметров времени τ_i и вероятности p_i на изменения времени выполнения запроса рассмотрим два