УДК 004.65

© 2004 г. **Н.А. Гребенников**,
Ю.А. Григорьев, д-р техн. наук
(Московский государственный технический университет им. Н.Э.Баумана),
А.Д. Плутенко, канд. техн. наук
(Амурский государственный университет, Благовещенск)

АНАЛИЗ РАЗМЕРОВ ПРОСТРАНСТВА ПОИСКА ОПТИМАЛЬНЫХ ПЛАНОВ ВЫПОЛНЕНИЯ ЗАПРОСОВ В СУБД

Анализируется задача оценки размеров пространства поиска, которое просматривается СУБД при оптимизации запросов, включающих соединения таблиц базы данных. В центре внимания – различные типы деревьев поиска и топологий исходных запросов. Оценивается выигрыш от использования memo-структуры и мультивыражений для представления деревьев поиска в пространстве поиска.

Введение

Системы управления базами данных (СУБД) являются основным компонентом современных систем обработки информации. С целью уменьшения времени выполнения запросов в состав СУБД включается оптимизатор, задача которого – анализ множества альтернативных планов выполнения запросов и выбор оптимального плана. Время поиска оптимального плана определяется размером пространства поиска, которое, в свою очередь, зависит от количества альтернативных деревьев поиска для исходного запроса. Поэтому представляется важной оценка объема вычислений для различных типов деревьев поиска и топологий запросов.

Деревья поиска и топологии запросов

Количество альтернативных деревьев поиска для запросов, содержащих только операторы соединения и доступа к файлам, представляет собой число альтернативных порядков соединения, или деревьев соединений. *Дерево соединений* – это регулярное бинарное дерево, «листья» его представляют собой базовые таблицы, которые указаны в исходном запросе, а промежуточные узлы моделируют операторы соединения. Эти операторы получают входные таблицы через входные дуги и пересылают таблицу результата операции по выходной дуге к следующему оператору. «Корень» (вершина) дерева определяет результат всего запроса. Дерево соединений является бинарным, так как реляционный оператор соединения имеет два входа.

При поиске оптимального плана оптимизатор обычно строит деревья поиска одного из двух типов: лево-глубокие и упорядоченные кустовые (рис. 1). *Лево-глубокое* (левостороннее, левонаправленное, леволинейное, линейное) дерево поиска – это дерево поиска, в котором для каждого оператора соединения (промежуточного узла) хотя бы один из входов является базовой таблицей; в противном случае дерево соединений называется *кустовым*.

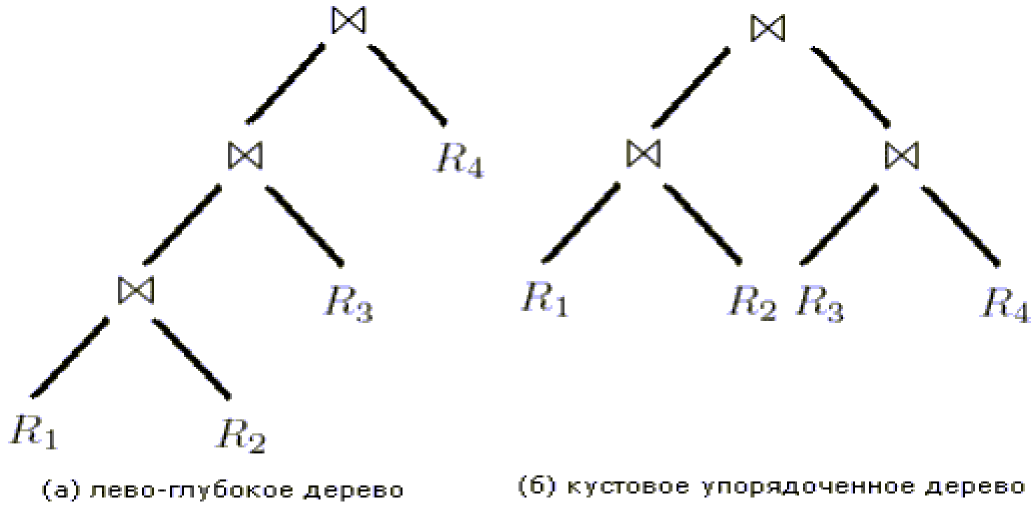

Рис. 1. Типы деревьев поиска.

Кроме типа деревьев поиска, на построение которых настроен оптимизатор, размер генерируемого им пространства поиска зависит также от топологии графа исходного запроса (далее просто *топологии запроса*). Существуют три типовые топологии запросов: линейная, звездообразная и полностью соединенная (рис. 2а).

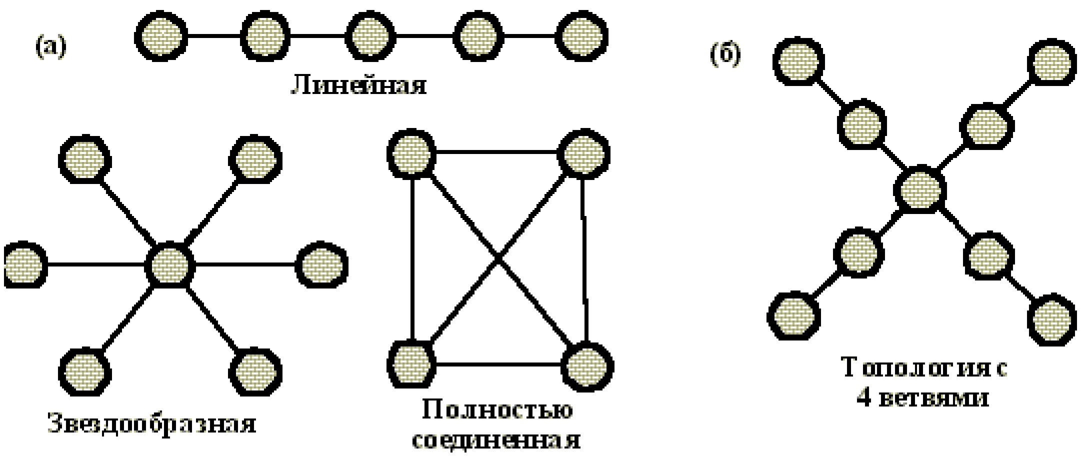

Рис. 2. Типы топологий запросов.

Линейный (строковый) запрос для *n* отношений состоит из двух терминальных (крайних) отношений, каждое из которых соединено только с одним отношением, и *n*-2 внутренних отношений, каждое из которых соединено с двумя другими соседними отношениями. Связь между двумя отношениями означает наличие в исходном запросе предиката соединения для этих отношений. *Звездообразный* запрос для n отношений состоит из одного центрального отношения и n-1 терминальных отношений, которые соединены только с центральным. *Полностью соединенный* запрос для n отношений состоит из n отношений, каждое из которых соединено со всеми другими n-1 отношениями. Любой запрос можно представить в виде полностью соединенного запроса, достроив недостающие связи путем добавления фиктивного предиката соединения, который всегда имеет значение «истина».

Добавленные таким образом связи, очевидно, будут представлять декартовы произведения соответствующих отношений.

Линейная и звездообразная топологии запросов представляют собой крайние случаи *ациклических* запросов [1]. Так, линейный запрос содержит одну ветвь, а звездообразный – n ветвей графа запроса соединений n таблиц. В работе [1] показано, что количество альтернативных деревьев поиска для всех остальных топологий ациклических запросов (с 2, 3 и т.д. ветвями при фиксированном числе отношений, см. рис. 2б) лежит в интервале от числа деревьев поиска для линейных запросов до числа деревьев поиска для звездообразных запросов.

Оценка числа альтернативных деревьев поиска

Найдем число возможных деревьев поиска для указанных топологий запросов и типов деревьев поиска. При этом будем рассматривать только логические деревья поиска, не учитывая различные физические реализации операции соединения.

Случай 1. Кустовое дерево, полностью соединенный запрос.

Этот случай дает верхнюю границу числа возможных деревьев поиска. В работе [2] приведена оценка числа деревьев поиска для этого случая:

$$x_n = \frac{(2n-2)!}{(n-1)!}. \qquad (1)$$

Предполагается, что для каждого дерева поиска корневой узел имеет левый операнд, охватывающий k отношений, и правый операнд, охватывающий (n-k) отношений. Поэтому число деревьев поиска определяется с помощью следующего выражения:

$$x_n = \sum_{k=1}^{n-1} \binom{n}{k} x_k x_{n-k}, \textit{ где } x_1 = 1. \qquad (2)$$

Далее путем достаточно сложных математических вычислений из (2) выводится искомая формула (1). В настоящей статье предлагается более простой вывод формулы (1) для числа деревьев поиска в рассматриваемом случае.

Из курса дискретной математики известно (например, [3]), что одной из интерпретаций *последовательности Каталана*[1] является число бинарных деревьев поиска с n+1 листьями. Отсюда число бинарных деревьев поиска с n «листьями» определяется (n-1)-м числом Каталана. Так как «листья» в рассматриваемом дереве поиска не упорядочены, можно менять их порядок любым способом для заданного дерева поиска. Это дает n! различных расположений «листьев». Поэтому получим следующее число деревьев поиска:

$$n!*C_{n-1} \overset{(3*)}{=} \frac{n!}{(n+1-1)}*\binom{2n-2}{n-1} = \frac{(n-1)!(2n-2)!}{(n-1)!(2n-2-n+1)!} = \frac{(2n-2)!}{(n-1)!}, \quad (3)$$

$$C_n = \frac{1}{(n+1)}*\binom{2n}{n}, \quad (3*)$$

Случай 2. Кустовое дерево, линейный запрос.

Так как линейная цепочка может быть разбита в k-м месте и любая часть может быть левым входом, то число деревьев поиска в этом случае:

$$y_n = \sum_{k=1}^{n-1} 2y_k y_{n-k}, \quad \text{где} \quad y_1 = 1. \quad (4)$$

В работе [2] показано, что имеет место следующее соотношение:

$$y_n = 2^{n-1} x_n \big/ n!.$$

Таким образом, число деревьев поиска в рассматриваемом случае может быть оценено формулой:

$$y_n = \frac{2^{n-1}}{n!}\frac{(2n-2)!}{(n-1)!}. \quad (5)$$

Этот случай дает нижнюю границу числа возможных деревьев поиска в классе ациклических запросов (см. рис. 2б).

Случай 3. Кустовое дерево, звездообразный запрос.

Данный случай на практике невозможен, так как по запросу звездообразной топологии невозможно построить кустовое дерево. Действитель-

---

[1] Последовательность Каталана (ПК) – последовательность вида (1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, 58786, 208012, 742900, 2674440, 9694845, ...), названная по имени Юджина Чарльза Каталана (1814 - 1894). Встречается в самых разных ситуациях, – например, определяет число правильных скобочных структур длины $2n$. ПК определяется следующей рекуррентной формулой: $k_n = \sum_{m=0}^{n-1}\binom{n}{m}k_m k_{n-m-1}$. Производящая функция $K_x = k_0 + k_1 x + k_2 x^2 + \ldots$ последовательности Каталана удовлетворяет условию $xK_x^2 - K_x + 1 = 0$. Решая это уравнении и используя условие K(0)=1, получаем функцию $K_x = \frac{1-\sqrt{1-4x}}{2x}$. Разложив функцию $\sqrt{1-4x}$ в ряд, получаем выражение для k_n: $k_n = \frac{1}{(n+1)}*\binom{2n}{n}$.

но, тогда необходимо, чтобы два входящих в соединение отношения разделяли центральную таблицу, но входные отношения для реляционной операции соединения не должны пересекаться.

Случай 4. Лево-глубокое дерево, полностью соединенный запрос.

Этот случай достаточно тривиален. Из определения лево-глубокого дерева видно, что дерево определяется упорядоченной последовательностью «листьев» (базовых отношений, присоединяемых на каждом шаге соединения). Число различных способов, которыми может быть упорядочено множество, состоящее из n элементов, равно числу перестановок элементов и выражается как $n!$ (при этом на каждом шаге упорядочивания не существует никаких ограничений для выбора следующего элемента, так как запрос является полностью соединенным).

Случай 5. Лево-глубокое дерево, линейный запрос.

Существует n-1 вариант для выбора первого оператора соединения в дереве поиска, при этом число операторов k, которые будут расположены слева от данного оператора, будет лежать в диапазоне [0…n-2] (для запроса соединения n отношений существует всего n-1 оператор соединения, плюс один оператор уже выбран). После этого будет существовать

$$\binom{n-2}{k}$$

способов достроить план (число вариантов переходов от правых таблиц к левым относительно первого оператора соединения). Следовательно, общее число вариантов будет равно

$$\sum_{k=0}^{n-2}\binom{n-2}{k}.$$

Так как для операции соединения важен порядок операндов, то для получения числа деревьев поиска в рассматриваемом случае число вариантов следует умножить на два, т.е.

$$z_n = 2*\sum_{k=0}^{n-2}\binom{n-2}{k} = 2*2^{n-2} = 2^{n-1}. \quad (6)$$

Случай 6. Лево-глубокое дерево, звездообразный запрос.

Как указано выше, дерево определяется упорядоченной последовательностью «листьев». Первым оператором соединения в дереве поиска для звездообразного запроса будет оператор соединения одного из терминальных отношений с центральным.

Следовательно, существуют два варианта для выбора первых двух «листьев» в дереве поиска: в качестве первого «листа» можно выбрать центральное(одним способом) либо терминальное (n-1 способом) отношение. Третий и последующий «листы» можно выбрать из любых оставшихся терминальных отношений, т.е. n-2, n-3, …, 2, 1 способами.

Таким образом, общее число деревьев поиска в рассматриваемом случае может быть оценено как[2]:

$$z_n=1*(n-1)*(n-2)*\ldots*2*1+(n-1)*1*(n-2)*\ldots*2*1=2(n-1)!. \qquad (7)$$

Полученные в данном разделе результаты оценки числа альтернативных деревьев поиска сведены в табл. 1.

Таблица 1

Топология запроса/Тип дерева	Кустовое дерево	Лево-глубокое дерево
Полностью соединенный	$\frac{(2n-2)!}{(n-1)!}$	$n!$
Линейный	$\frac{2^{n-1}}{n!}\frac{(2n-2)!}{(n-1)!}$	2^{n-1}
Звездообразный	-	$2(n-1)!$

На основе данных, приведенных в табл. 1, на рис. 3 представлены графики числа деревьев поиска для запросов, включающих от 3 до 13 соединяемых отношений (использована логарифмическая шкала).

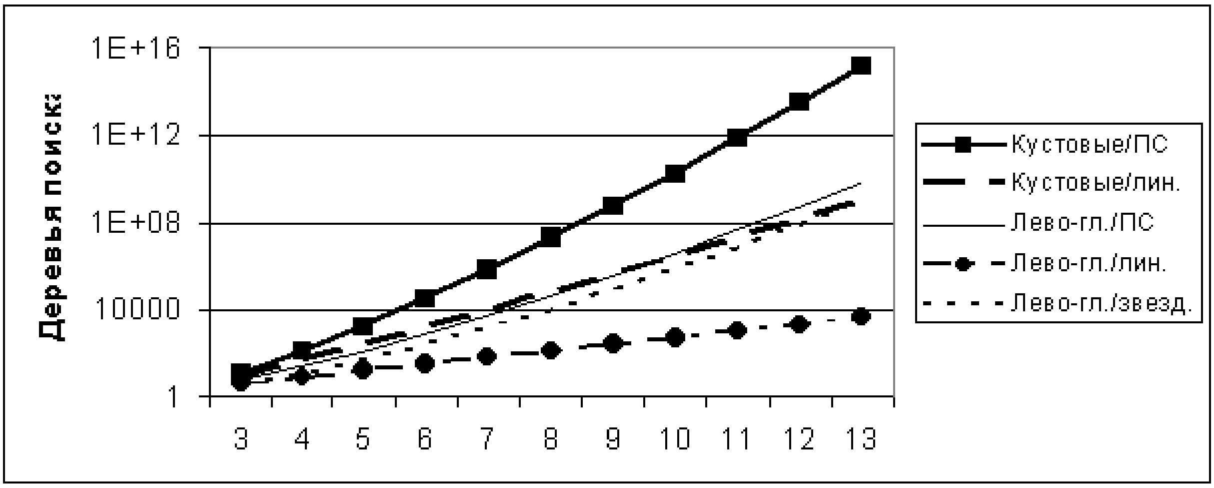

Рис. 3. Число деревьев поиска для запросов соединений.

Оценка числа мультивыражений в memo-структуре

В предыдущем разделе показано, что при поиске оптимального плана выполнения запросов появляется проблема комбинаторной сложности, возникающая из-за быстрого роста числа альтернатив. Так, для полностью соединенного запроса с 7 отношениями существует 665280 альтернативных кустовых деревьев поиска. Применение эвристики генерации только лево-глубоких деревьев поиска позволяет снизить данное число до 5040.

Для решения этой проблемы в оптимизаторах Volcano [4] и Cascades [5] используется специальная структура данных (memo-структура). Основ-

[2] Этот же результат можно получить, заметив, что для звездообразного запроса каждое из n-1 терминальных отношений связано с остальными n-2 терминальными отношениями через центральное отношение; после выполнения первой же операции соединения запрос становится «как бы» полностью соединенным запросом соединения, но уже для n-1 отношения. Используя полученный ранее результат (случай 4), получаем формулу $2(n-1)!$

ная идея использования memo-структуры состоит в том, чтобы избежать повторного рассмотрения поддеревьев, сохраняя ранее рассмотренные разделяемые копии. Выигрыш от такого подхода можно показать с помощью дерева рекурсии, представленного на рис. 4.

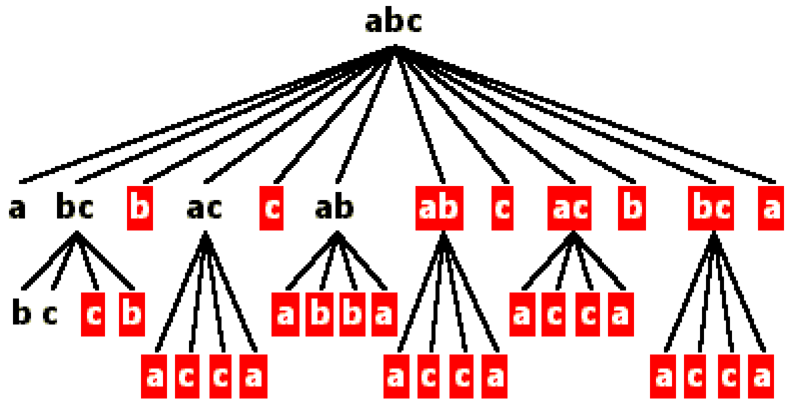

Рис. 4. Дерево рекурсии для запроса соединения трех отношений.

Это дерево вызовов функции оптимизации для промежуточных результатов (поддеревьев), выполняемых при построении всех альтернативных деревьев поиска. На рис. 3 выделены те вершины, в которых происходят повторные обращения к уже рассмотренным (оптимизированным) поддеревьям. Видно, что число повторений даже для представленного простого запроса соединения трех отношений очень велико.

Как указано в работе [5], memo-структура организована как сеть групп (классов эквивалентности). Каждая группа состоит из набора мультивыражений, которые генерируют одинаковый (промежуточный) результат. Входами мультивыражений являются группы, что означает – «любое мультивыражение группы может быть использовано в качестве входа».

Для определения размеров memo-структуры оценим число мультивыражений, которое окажется в пространстве поиска после окончания процесса перебора. Как и при оценке альтернативных деревьев поиска, рассмотрим два типа генерируемых деревьев поиска и три типа топологии запросов.

Случай 1. Кустовые деревья поиска.

Теорема 1. Максимальное количество мультивыражений (операторов соединения), необходимых для представления всех кустовых деревьев соединения для полностью соединенных запросов с числом отношений n равно

$$3^n\text{-}2^{n+1}+n+1.$$

Доказательство. Оценим сначала число групп в пространстве поиска. Так как каждое непустое подмножество множества базовых отношений представляет собой некоторой промежуточный результат, то число групп в пространстве поиска равно

$$\sum_{k=1}^{n}\binom{n}{k}=2^n-1.$$

Группа, соответствующая k базовым отношениям, описывает все возможные операторы верхнего уровня для этих k отношений. Каждое разделение на левое/правое непустые подмножества множества k отношений соответствует оператору в группе. Таким образом, число операторов в группе, соответствующей k базовым отношениям, равно 2^k-2 для $k>1$. При $k=1$ группа содержит единственный оператор доступа к базовому отношению. По условию существует n таких групп. Число операторов в memo-структуре есть сумма всех операторов во всех группах memo-структуры

$$\sum_{k=2}^{n}\binom{n}{k}(2^k-2)+n=\sum_{k=2}^{n}\binom{n}{k}*2^k-2\sum_{k=2}^{n}\binom{n}{k}+n\overset{(8^*,8^{**})}{=} \tag{8}$$

$$=3^n-1-2n-2*2^n+2+2n+n=3^n-2^{n+1}+n+1.$$

Бином Ньютона (основная биноминальная теорема) (8*)

$$\sum_{k=2}^{n}x=\sum_{k=0}^{n}x-x\big|_0-x\big|_1. \tag{8**}$$

<u>Теорема 2.</u> Количество мультивыражений (операторов соединения), необходимых для представления всех кустовых деревьев соединения для линейных запросов с числом отношений n, равно

$$\frac{n^3+2n}{3}.$$

<u>Доказательство.</u> Для строкового запроса, выполняющего соединение n отношений, существует n-k+1 возможных «подстрок», связывающих k базовых отношений. Для каждой их этих подстрок существует группа, содержащая все операторы верхнего уровня этой группы. Если левый и правый входы различаются, то имеется 2*(k-1) операторов в группе, которая описывает подстроку из k отношений для $k>1$. При $k=1$ группа содержит единственный оператор. Таким образом, общее число операторов в memo-структуре равно

$$2\sum_{k=2}^{n}(n-k+1)(k-1)+n=2\sum_{k=2}^{n}(-n-1-k^2+k(n+2))+n\overset{(9^*,9^{**})}{=}$$

$$=-2(n-1)(n+1)-2\left(\frac{2n^3+3n^2+n}{6}-1\right)+2(n+2)\left(\frac{n^2+n}{2}-1\right)+n= \tag{9}$$

$$=-2n^2+2-\frac{2n^3+3n^2+n-6}{3}+n^3+3n^2-4+n=\frac{n^3+2n}{3},$$

$$\sum_{k=2}^{n}x=\sum_{k=1}^{n}x-x\big|_1, \tag{9*}$$

$$\sum_{k=1}^{n}x=nx,\ \sum_{k=1}^{n}k=\frac{n(n+1)}{2},\ \sum_{k=1}^{n}k^2=\frac{n(n+1)(2n+1)}{6}, \tag{9**}$$

что следует из работы [6].

Случай 2. Лево-глубокие деревья поиска.

Теорема 3. Количество мультивыражений (операторов соединения), необходимых для представления всех лево-глубоких деревьев соединения для полностью соединенных запросов с числом отношений n, равно $n2^{n-1}$.

Доказательство. Число групп, соответствующих k базовым отношениям, для полностью соединенных запросов определяется как

$$\binom{n}{k}.$$

Число мультивыражений в группе, соответствующих k базовым отношениям, равно k, так как для каждого мультивыражения в группе правый вход является базовым отношением. При k=1 группа содержит единственный оператор доступа к базовому отношению. Таким образом, общее число операторов в memo-структуре равно

$$\sum_{k=2}^{n}\binom{n}{k}k+n \overset{(10^*)}{=} n*2^{n-1}-n+n=n*2^{n-1}, \qquad (10)$$

$$\sum_{k=0}^{n}\binom{n}{k}k=\sum_{k=0}^{n}\frac{k*n!}{k!*(n-k)!}=\sum_{k=0}^{n}n*\frac{(n-1)!}{(k-1)!*(n-1-(k-1))!}=n*2^{n-1}. \qquad (10^*)$$

Теорема 4. Количество мультивыражений (операторов соединения), необходимых для представления всех лево-глубоких деревьев соединения для линейных запросов с числом отношений n, равно n^2 .

Доказательство. Для строкового запроса, выполняющего соединение n отношений, существует n-k+1 возможных «подстрок», связывающих k базовых отношений. Для каждой их этих подстрок существует группа, содержащая все операторы верхнего уровня этой группы. В каждой группе, затрагивающей два и более отношения, существуют всего два мультивыражения, так как для каждого мультивыражения в группе правый вход должен являться базовым отношением и для каждой «подстроки» размера k>1 существует только два терминальных отношения, которые могут быть правым входом. При k=1 каждая группа содержит единственный оператор доступа к базовому отношению. Таким образом, общее число операторов в memo-структуре равно

$$\sum_{k=2}^{n}2(n-k+1)+n \overset{9^*,9^{**}}{=} 2(n+1)(n-1)-2\left(\frac{n^2+n}{2}-1\right)+n=n^2. \qquad (11)$$

Теорема 5. Количество мультивыражений (операторов соединения), необходимых для представления всех лево-глубоких деревьев соединения для звездообразных запросов с числом отношений n, равно

$$(n-1)2^{n-2}+2n-1.$$

Доказательство. Число групп, которые связывают k базовых отношений, для звездообразного запроса определяется как

$$\binom{n-1}{k-1},$$

так как центральное отношение включается во все соединения и, следовательно, во все группы, а оставшиеся k-1 отношения выбираются из n-1 терминальных базовых отношений. В каждой группе, где соединяются три и более базовых отношения, существует k-1 мультивыражений, так как для каждого мультивыражения в группе правый вход должен являться базовым отношением и центральное отношение не может быть правым входом, иначе левый вход будет представлять собой совокупность декартовых произведений. Таким образом, каждое из k-1 терминальных отношений используется в качестве правого входа одного из мультивыражений группы. В каждой группе, которая связывает два базовых отношения, существуют два мультивыражения, так как эти группы представляют оператор соединения нижнего уровня в дереве поиска, для которого перестановка входов не нарушает структуры лево-глубокого дерева.

Таким образом, общее число операторов в memo-структуре равно

$$\begin{aligned}
&\sum_{k=3}^{n}\binom{n-1}{k-1}(k-1)+2\binom{n-1}{1}+n \overset{(4,3)}{=} \\
&=\sum_{k=0}^{n}\binom{n-1}{k-1}-\left.\binom{n-1}{k-1}\right|_0-\left.\binom{n-1}{k-1}\right|_1-\left.\binom{n-1}{k-1}\right|_2+2(n-1)+n= \\
&=(n-1)\sum_{k=0}^{n}\frac{(n-2)!}{(k-2)!(n-2-(k-2))!}-0-0-(n-1)+3n-2= \\
&=(n-1)*2^{n-2}+2n-1.
\end{aligned} \tag{12}$$

Доказательство теоремы 5 завершено.

В работе [1] также рассматривались указанные типы деревьев поиска и топологии запросов для оценки числа операторов соединения, генерируемых оптимизатором Starburst. Полученные в настоящей работе результаты соотносятся с результатами, представленными в работе [1], однако существует различие, обусловленное тем, что в настоящей работе рассматриваются упорядоченные деревья соединения и учитывается n операторов для доступа к базовым отношениям.

Полученные в данном разделе результаты оценки числа мультивыражений в memo-структуре представлены в табл. 2.

Таблица 2

Топология запроса/Тип дерева	Кустовое дерево	Лево-глубокое дерево
Полностью соединенный	$3^n-2^{n+1}+n+1$	$n2^{n-1}$
Линейный	$\frac{n^3+2n}{3}$	n^2
Звездообразный	–	$(n-1)2^{n-2}+2n-1$

На основе данных, представленных в табл. 2, на рис. 5 приведены графики числа мультивыражений в memo-структуре, которые необходимы для представления всех альтернативных деревьев поиска для запросов, включающих от 3 до 13 соединяемых отношений (использована логарифмическая шкала):

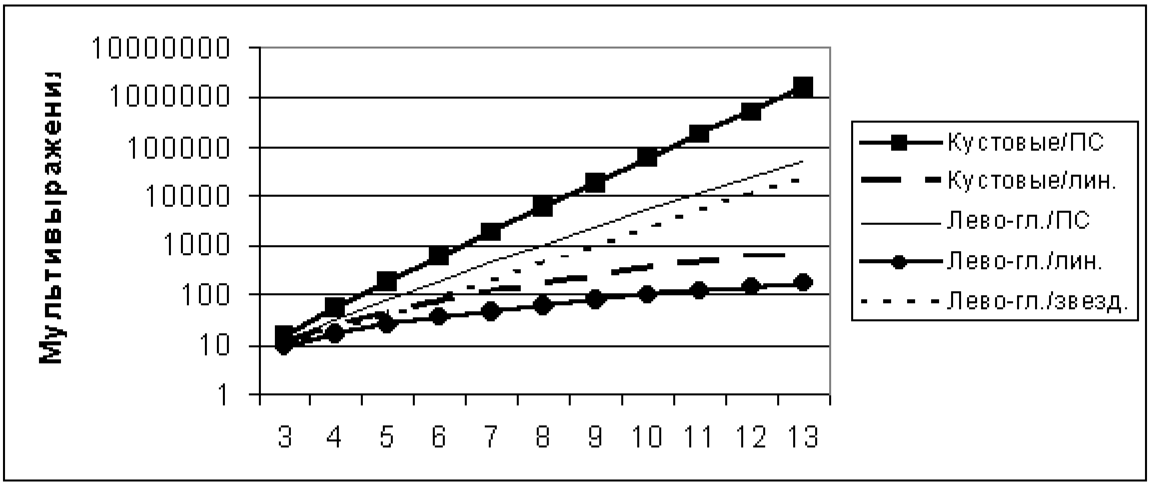

Рис. 5. Число мультивыражений для запросов соединений.

Оценка эффективности использования memo-структуры для представления пространства поиска

Полученные выше результаты позволяют выполнить оценку эффективности использования memo-структуры для представления деревьев соединений в пространстве поиска. Хотя в предыдущих разделах получены результаты для объектов различного типа (деревьев поиска и мультивыражений), возможность сравнения обусловлена тем, что оба типа объектов представляются в процессе работы оптимизатора похожими структурами данных (программными единицами), одинаковым образом требующих расхода оперативной памяти и других ресурсов компьютера.

В табл. 3 представлено количество альтернативных деревьев поиска и необходимых для их представления мультивыражений в memo-структуре для запросов, включающих от 3 до 7 соединяемых отношений, для различных типов деревьев поиска и топологий запросов.

Таблица 3

Число отнош.	Кустовые деревья				Лево-глубокие деревья					
	полн. соед.		линейные		полн. соед.		линейные		«звезда»	
	дерев.	опер.	дерев.	опер.	дерев.	опер.	дерев.	опер.	дерев.	опер.
3	12	15	8	11	6	12	4	9	4	9
4	120	54	40	24	24	32	8	16	12	19
5	1680	185	224	45	120	80	16	25	48	41
6	30240	608	1344	76	720	192	32	36	240	91
7	665280	1939	8448	119	5040	448	64	49	1440	272

Выигрыш, получаемый за счет использования memo-структуры и мультивыражений, для конкретного типа конечного дерева поиска и топологии исходного запроса можно оценить, сравнив значение в соответствующей ячейке столбца «дерев.» со значением ячейки столбца «опер.». Например, для случая кустового дерева поиска и полностью соединенного исходного запроса соединения 4 отношений существует 120 альтернативных деревьев поиска и требуется 54 мультивыражения (оператора соединения) для их представления. Следовательно, получаемый выигрыш равен (120-54)/120, т.е. 55%.

На рис. 6 показаны зависимости величины выигрыша от использования memo-структуры и мультивыражений для представления деревьев поиска для различных типов деревьев поиска и топологий исходного запроса для запросов, включающих от 3 до 13 соединяемых отношений.

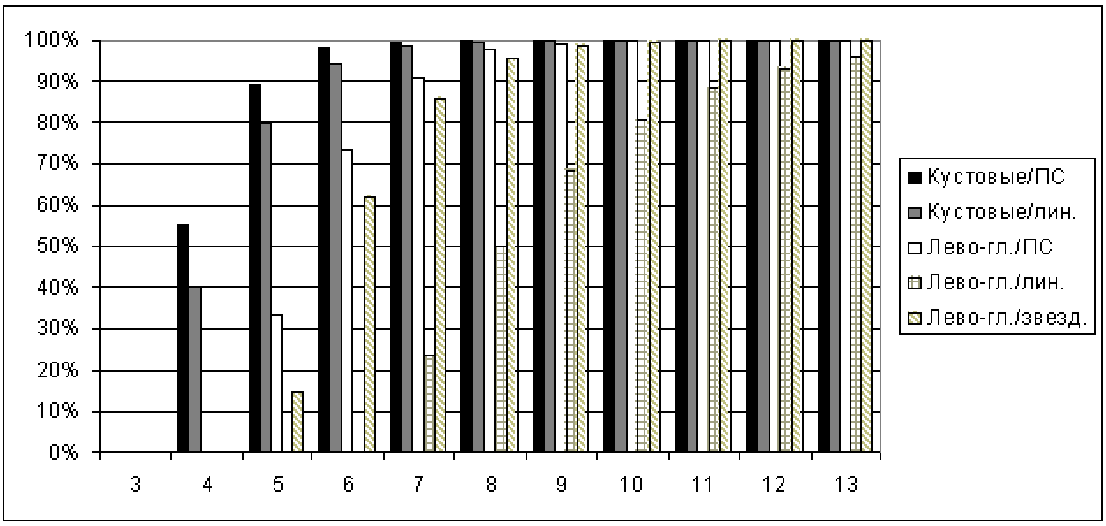

Рис. 6. Выигрыш от использования memo-структуры и мультивыражений для представления деревьев поиска в пространстве поиска.

Из рис. 6 видно, что наибольший выигрыш достигается для кустовых деревьев поиска, для которых при любой топологии исходного запроса соединения 6 отношений выигрыш составляет более 95%. Выигрыш, достигаемый для лево-глубоких деревьев поиска, менее значителен, однако и в этом случае для полностью соединенных и звездообразных запросов выигрыш приближается к 90%, начиная с 7-8-го отношений. Медленнее всего растет величина выигрыша для линейных запросов при построении лево-глубоких деревьев поиска. Это объясняется малым числом деревьев для данного случая, поэтому выигрыш от использования memo-структуры начинает ощущаться только для запросов соединения 6 отношений и достигает величины 90% только для запросов с 11 отношениями. Отсутствие выигрыша в некоторых случаях для запросов, включающих малое число отношений (от 3 до 5), объясняется малым числом деревьев поиска, а также учетом при подсчете числа мультивыражений *n* операторов доступа к базовым отношениям, которые не являются операторами соединения.

Таким образом, можно сделать вывод, что использование memo-структуры и мультивыражений для представления деревьев поиска в пространстве поиска позволяет получить значительный выигрыш как в оперативной памяти, так и в быстродействии работы оптимизатора запросов СУБД.

Заключение

Получены формулы для оценки числа альтернативных деревьев поиска и необходимых для их представления мультивыражений в memo-структуре для кустовых и лево-глубоких деревьев поиска и полностью соединенных, линейных и звездообразных топологий исходных запросов.

На основе полученных формул проведена оценка эффективности использования memo-структуры для представления пространства поиска, показавшая, что использование memo-структуры и мультивыражений позволяет получить значительный выигрыш в оперативной памяти и быстродействии работы оптимизатора СУБД для запросов соединений, включающих более 5 таблиц базы данных.

ЛИТЕРАТУРА

1. *Ono K., Lohman G.* Measuring the Complexity of Join Enumeration in Query Optimization. // Proceedings of the 16th International Conference on Very Large Data Bases. Brisbane, Australia. August 1990. P.314-325.
2. *Lanzelotte R., Valduriez P., Zait M.* On the effectiveness of optimization search strategies for parallel execution spaces // Proc. of the 19th VLDB Conference. Dublin, 1993. P.493-504.
3. *Hilton P., Pedersen J.* Catalan numbers, their generalizations and their uses // The Mathematical Intelligencer, 13 (1991). P.64-75.
4. *Graefe G., McKenna W.* The Volcano Optimizer Generator: Extensibility and Efficient Search. In Proceeding of the 12-th International Conf. on Data Engineering, 1993. P.209-218.
5. *Graefe G.* The Cascades Framework for Query Optimization // Bulletin of the IEEE Technical Committee on Data Engineering, 18(3), September 1995. P.19-29.
6. *Корн Г., Корн Т.* Справочник по математике. М.: Наука, 1984.