УДК 004.65

Ю. А. Григорьев, Н. А. Гребенников

ОЦЕНКА РАЗМЕРОВ ПРОСТРАНСТВА ПОИСКА В ОПТИМИЗАТОРАХ ЗАПРОСОВ СИСТЕМ УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ

Рассмотрена задача оценки размера пространства поиска, которое просматривается системами управления базами данных при оптимизации запросов, включающих соединения таблиц базы данных. Проанализированы различные типы деревьев поиска и топологий исходных запросов. Оценена величина выигрыша от использования memo-структуры и мультивыражений для представления деревьев поиска в пространстве поиска.

Системы управления базами данных (СУБД) являются основным компонентом современных систем обработки информации. С целью уменьшения времени выполнения запросов в состав СУБД включается оптимизатор запросов, задачей которого является 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 задача оценки 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" которого представляют собой базовые таблицы, указанные в исходном запросе, а промежуточными узлами моделируются операторы соединения. Эти операторы получают со входа таблицы через входные дуги и пересылают таблицу результата операции по выходной дуге к следующему оператору. "Корень" дерева определяет результат всего запроса. Дерево соединений является бинарным, так как реляционный оператор соединения имеет два входа.

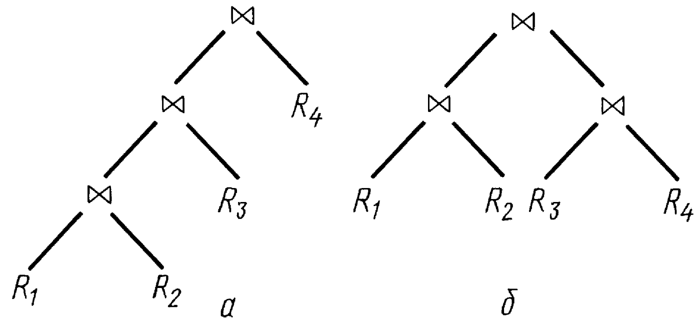

Рис. 1. Типы деревьев поиска:
а — левоглубокое дерево; *б* — кустовое упорядоченное дерево

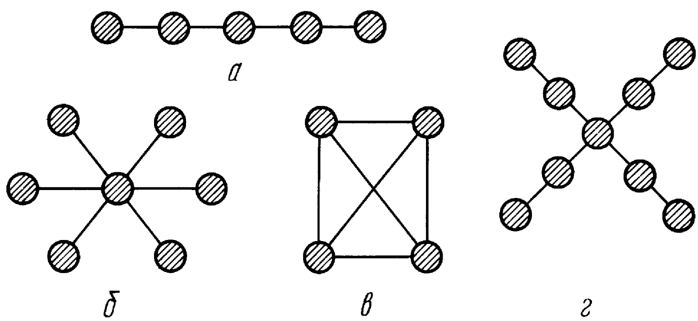

Рис. 2. Типы топологий запросов:
а — линейная; *б* — звездообразная; *в* — полностью соединенная; *г* — с четырьмя ветвями

При поиске оптимального плана оптимизатор обычно строит деревья поиска одного из двух типов (рис. 1). Левоглубокое (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.

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 (рис. 2, *а*–*в*). 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 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 собой декартовы произведения соответствующих отношений.

Линейная и звездообразная топологии запросов представляют собой крайние случаи ациклических запросов [1]. 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 двумя, тремя и т.д. ветвями при фиксированном числе отношений — рис. 2, *г*) 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 реализации операции соединения.

Случай 1. Кустовое дерево, полностью соединенный запрос. Этот случай определяет верхнюю границу числа возможных деревьев поиска. В работе [2] дана оценка числа деревьев поиска для этого случая:

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

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:

$$x_n = \sum_{k=1}^{n-1} \binom{n}{k} x_k x_{n-k}, \tag{2}$$

где $x_1 = 1$.

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

Из работ по дискретной математике (например, из работы [3]) известно, что одной из интерпретаций последовательности Каталана[1] является число бинарных деревьев поиска с $n+1$ "листьями". Таким образом, число бинарных деревьев поиска с n "листьями" определяется $(n-1)$-м числом последовательности Каталана. Поскольку "листья" в рассматриваемом дереве поиска не упорядочены, то можно менять их порядок любым способом для заданного дерева поиска. Следовательно, существует $n!$ различных расположений "листьев". Поэтому получим следующее число деревьев поиска:

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

$$C_n = \frac{1}{(n+1)}\binom{2n}{n}.$$

Случай 2. Кустовое дерево, линейный запрос. Поскольку линейная цепочка может быть разорвана на k-й позиции (на k-м отношении) и любая ее часть может быть левым входом, то число деревьев поиска в этом случае описывается выражением

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

где $y_1 = 1$.

В работе [2] показано, что справедливо следующее соотношение: $y_n = 2^{n-1}x_n/n!$. Таким образом, число деревьев поиска в рассматриваемом случае может быть оценено формулой

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

[1]Последовательность Каталана — 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). Последовательность Каталана, например, определяет число правильных скобочных структур длины $2n$ и определяется следующей рекуррентной формулой: $k_n = \sum_{m=0}^{n-1}\binom{n}{m}k_m k_{n-m-1}$. Производящая функция $K_x = k_0 + k_1x + k_2x^2 + \ldots$ последовательности Каталана удовлетворяет условию $xK_x^2 - K_x + 1 = 0$. Решая это уравнение 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: $k_n = \frac{1}{n+1}\binom{2n}{n}$.

Этот случай определяет нижнюю границу числа возможных деревьев поиска в классе ациклических запросов (см. рис. 2, *г*).

Случай 3. Кустовое дерево, звездообразный запрос. Данный случай на практике невозможен, так как по запросу звездообразной топологии невозможно построить кустовое дерево, поскольку необходимо, чтобы два входящих в соединение отношения разделяли центральную таблицу, но входные отношения для реляционной операции соединения не должны пересекаться.

Случай 4. Левоглубокое дерево, полностью соединенный запрос. Этот случай достаточно тривиален. Из определения левоглубокого дерева следует, 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 упорядочения не существует никаких ограничений для выбора следующего элемента, так как запрос является полностью соединенным).

Случай 5. Левоглубокое дерево, линейный запрос. Существует $n-1$ вариант для выбора первого оператора соединения в дереве поиска, при этом число операторов k, которые расположены слева от данного оператора, лежит в интервале $0\ldots n-2$ (для запроса соединения n отношений всего существует $n-1$ оператор соединения, при этом один оператор уже выбран). Существует $\binom{n-2}{k}$ способов достроить план (число вариантов переходов от правых таблиц к левым относительно первого оператора соединения). Следовательно, общее число вариантов будет равно $\sum\limits_{k=0}^{n-2}\binom{n-2}{k}$. Поскольку для операции соединения порядок операндов является важным, то для получения числа деревьев поиска в рассматриваемом случае число вариантов следует умножить на два:

$$z_n = 2\cdot\sum_{k=0}^{n-2}\binom{n-2}{k} = 2\cdot 2^{n-2} = 2^{n-1}. \tag{6}$$

Случай 6. Левоглубокое дерево, звездообразный запрос. Как указано выше, дерево определяется упорядоченной последовательностью "листьев". Первым оператором соединения в дереве поиска для звездообразного запроса является оператор соединения одного из терминальных отношений с центральным. Следовательно, существует два варианта для выбора первых двух "листьев" в дереве поиска: в качестве первого "листа" можно выбрать центральное отношение (одним способом) либо терминальное ($n-1$ способом). Третий и последующий "листы" можно выбрать из любых оставшихся терминальных отношений, т.е. $n-2,\ n-3, \ldots, 2, 1$ способом. Общее число деревьев поиска в рассматриваемом случае может быть оценено следующим образом[1]:

$$z_n = 1\cdot(n-1)\cdot(n-2)\cdot\ldots\cdot2\cdot1+(n-1)\cdot1\cdot(n-2)\cdot\ldots\cdot2\cdot1 = 2(n-1)!. \quad (7)$$

Результаты оценки числа альтернативных деревьев поиска приведены в табл. 1.

Таблица 1

Топология запроса	Тип дерева	
	кустовое	левоглубокое
Полностью соединенная	$\frac{(2n-2)!}{(n-1)!}$	$n!$
Линейная	$\frac{2^{n-1}}{n!}\frac{(2n-2)!}{(n-1)!}$	2^{n-1}
Звездообразная	—	$2(n-1)!$

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).

Оценка числа мультивыражений в memo-структуре. При поиске оптимального плана выполнения запросов появляется проблема комбинаторной сложности, возникающая из-за быстрого возрастания числа альтернатив. Например, для полностью соединенного запроса с семью отношениями существует 665280 альтернативных кустовых деревьев поиска. Применение эвристики генерации только левоглубоких деревьев поиска позволяет снизить это число до 5040.

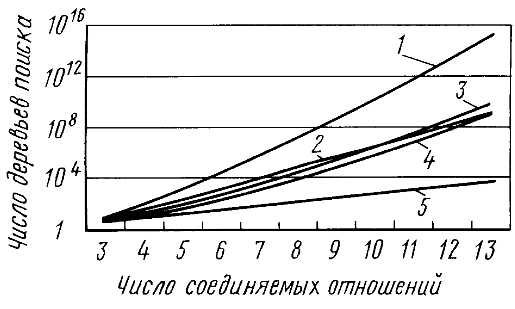

Рис. 3. Число деревьев поиска для запросов соединений:
1 — кустовое дерево, полностью соединенный запрос; *2* — кустовое дерево, линейный запрос; *3* — левоглубокое дерево, полностью соединенный запрос; *4* — левоглубокое дерево, звездообразный запрос; *5* — левоглубокое дерево, линейный запрос

[1]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, и после выполнения первой же операции соединения запрос становится полностью соединенным запросом соединения, но для $n-1$ отношения. Используя рассмотренный ранее результат (случай 4), получаем формулу $2(n-1)!$.

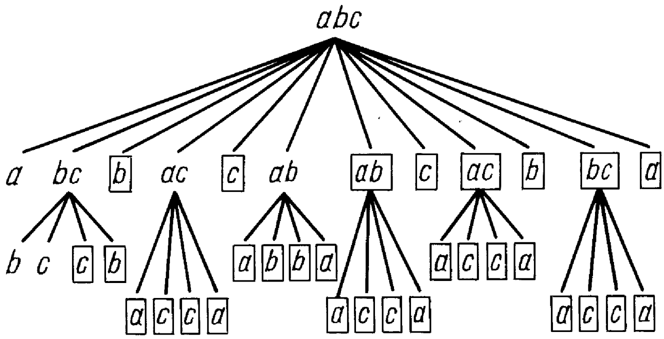

Рис. 4. Дерево рекурсии для запроса соединения трех отношений

Для решения этой проблемы в оптимизаторах Volcano [4] и Cascades [5] используется специальная структура данных (memo-структура). Основная 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. Дерево рекурсии является деревом вызовов функции, выполняемых при построении всех альтернативных деревьев поиска оптимизации, для промежуточных результатов (поддеревьев). На рис. 4 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 — это означает, что любое мультивыражение группы может быть использовано в качестве входа.

Для определения размеров memo-структуры оценим число мультивыражений, которые окажутся 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 $3^n - 2^{n+1} + n + 1$.*

Доказательство. Оценим сначала число групп в пространстве поиска. Поскольку каждое непустое подмножество множества базовых отношений представляет собой некоторый 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 и правое непустые подмножества множества k отношений соответствует оператору в группе. Таким образом, число операторов в группе, соответствующей k базовым отношениям, равно $2k - 2$ 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 — это число всех операторов во всех группах memo-структуры, следовательно, в силу бинома Ньютона имеем

$$\sum_{k=2}^{n} \binom{n}{k} (2^k - 2) + n = \sum_{k=2}^{n} \binom{n}{k} \cdot 2^k - 2 \sum_{k=2}^{n} \binom{n}{k} + n =$$

$$= 3^n - 1 - 2n - 2 \cdot 2^n + 2 + 2n + n = 3^n - 2^{n+1} + n + 1. \quad (8)$$

Теорема 2. *Количество мультивыражений (операторов соединения), необходимых для представления всех кустовых деревьев соединения, для линейных запросов с числом отношений* n *равно* $(n^3 + 2n)/3$.

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. 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, на основании выражений [6]

$$\sum_{k=1}^{n} k = \frac{n(n+1)}{2}, \quad \sum_{k=1}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}$$

получим, что общее число операторов в memo-структуре равно

$$2 \cdot \sum_{k=2}^{n} (n - k + 1)(k - 1) + n = 2 \cdot \sum_{k=2}^{n} (-n - 1 - k^2 + k(n+2)) + n =$$

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

$$= -2n^2 + 2 - \frac{2n^3 + 3n^2 + n - 6}{3} + n^3 + 3n^2 - 4 + n = \frac{n^3 + 2n}{3}. \quad (9)$$

Случай 2. Левоглубокие деревья поиска.

Теорема 3. *Количество мультивыражений (операторов соединения), необходимых для представления всех левоглубоких деревьев соединения, для полностью соединенных запросов с числом отношений n равно $2^{n-1}n$.*

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, на основании выражения

$$\sum_{k=0}^{n}\binom{n}{k}k = \sum_{k=0}^{n}\frac{kn!}{k!(n-k)!} = \sum_{k=0}^{n} n\frac{(n-1)!}{(k-1)!(n-1-(k-1))!} = 2^{n-1}n$$

получим, что общее число операторов в memo-структуре составляет

$$\sum_{k=2}^{n}\binom{n}{k}k + n = 2^{n-1}n - n + n = 2^{n-1}n. \tag{10}$$

Теорема 4. *Количество мультивыражений (операторов соединения), необходимых для представления всех левоглубоких 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, включающей два и более отношения, существует всего два мультивыражения, так как для каждого мультивыражения в группе правый вход является 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 составляет

$$\sum_{k=2}^{n}2(n-k+1) + n = 2(n+1)(n-1) - 2\left(\frac{n^2+n}{2} - 1\right) + n = n^2. \tag{11}$$

Теорема 5. *Количество мультивыражений (операторов соединения), необходимых для представления всех левоглубоких деревьев соединения, для звездообразных запросов с числом отношений n равно $2^{n-2}(n-1) + 2n - 1$.*

__Доказательство.__ Число групп, которые связывают k базовых отношений, для звездообразного запроса определяется как $\begin{pmatrix} n-1 \\ k-1 \end{pmatrix}$, поскольку 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 является 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, существует два мультивыражения, так как эти группы представляют собой оператор соединения нижнего уровня в дереве поиска, для которого перестановка входов не нарушает структуры левоглубокого дерева.

Таким образом, общее число операторов в memo-структуре в силу равенств (3) и (4) составляет

$$\sum_{k=3}^{n} \begin{pmatrix} n-1 \\ k-1 \end{pmatrix}(k-1) + 2\begin{pmatrix} n-1 \\ 1 \end{pmatrix} + n =$$

$$= \sum_{k=0}^{n} \begin{pmatrix} n-1 \\ k-1 \end{pmatrix} - \begin{pmatrix} n-1 \\ k-1 \end{pmatrix}\Bigg|_{k=0} -$$

$$- \begin{pmatrix} n-1 \\ k-1 \end{pmatrix}\Bigg|_{k=1} - \begin{pmatrix} n-1 \\ k-1 \end{pmatrix}\Bigg|_{k=2} + 2(n-1) + n =$$

$$= (n-1)\sum_{k=0}^{n} \frac{(n-2)!}{(k-2)!(n-2-(k-2))!} - (n-1) + 3n - 2 =$$

$$= 2^{n-2}(n-1) + 2n - 1. \qquad (12)$$

Доказательство теоремы 5 завершено.

В работе [1] рассмотрены 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 учитываются n операторов для доступа к базовым отношениям.

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

Таблица 2

Топология запроса	Тип дерева	
	кустовое	левоглубокое
Полностью соединенная	$3^n - 2^{n+1} + n + 1$	$2^{n-1}n$
Линейная	$\dfrac{n^3 + 2n}{3}$	n^2
Звездообразная	—	$2^{n-2}(n-1) + 2n - 1$

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).

Оценка эффективности использования memo-структуры для представления пространства поиска. Полученные выше результаты позволяют оценить эффективность использования memo-структуры для представления деревьев соединений в пространстве поиска. Несмотря на то, что 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), которые одинаковым образом требуют 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.

Выигрыш, получаемый за счет использования memo-структуры и мультивыражений, для конкретного типа конечного дерева поис-

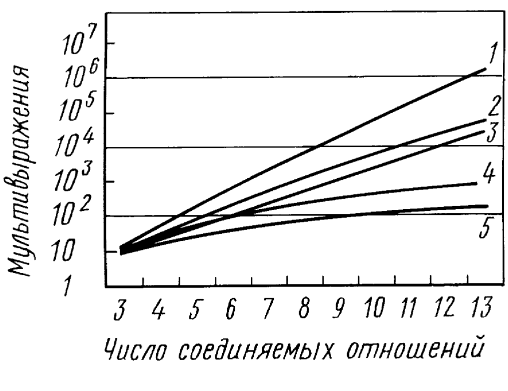

Рис. 5. Число мультивыражений для запросов соединений:
1 — кустовое дерево, полностью соединенный запрос; *2* — левоглубокое дерево, полностью соединенный запрос; *3* — левоглубокое дерево, звездообразный запрос; *4* — кустовое дерево, линейный запрос; *5* — левоглубокое дерево, линейный запрос

Таблица 3

Число отношений	Кустовые деревья				Левоглубокие деревья					
	Полностью соединенный запрос		Линейный запрос		Полностью соединенный запрос		Линейный запрос		Звездообразный запрос	
	Число деревьев	Число мульти-выражений	Число деревьев	Число мульти-выражений	Число деревьев	Число мульти-выражений	Число деревьев	Число мульти-выражений	Число деревьев	Число мульти-выражений
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

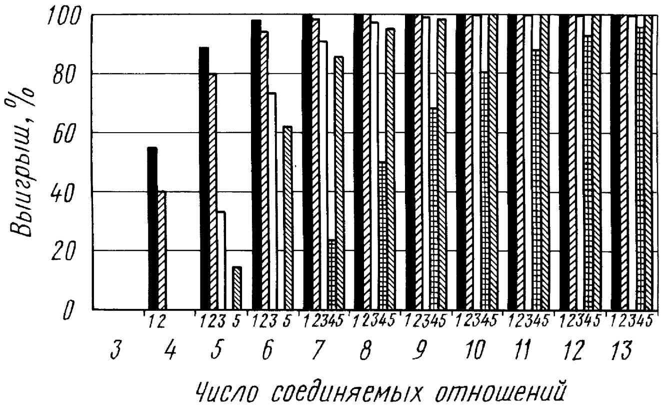

Рис. 6. Величина выигрыша от использования memo-структуры и мультивыражений для представления деревьев поиска в пространстве поиска:
1 — кустовое дерево, полностью соединенный запрос; *2* — кустовое дерево, линейный запрос; *3* — левоглубокое дерево, полностью соединенный запрос; *4* — левоглубокое дерево, линейный запрос; *5* — левоглубокое дерево, звездообразный запрос

ка и топологии исходного запроса можно оценить, сравнив значение, соответствующее числу деревьев, со значением, соответствующим числу мультивыражений. Например, для случая кустового дерева поиска и полностью соединенного исходного запроса соединения четырех 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 представлены диаграммы величин выигрыша для запросов, включающих от 3 до 13 соединяемых отношений. Из рисунка видно, что наибольший выигрыш достигается для кустовых деревьев поиска, для которых при любой топологии исходного запроса соединения шести отношений выигрыш составляет более $95\,\%$. Выигрыш для левоглубоких деревьев поиска менее значителен, однако и в этом случае для полностью соединенных и звездообразных запросов выигрыш приближается к $90\,\%$ начиная с 7–8 отношений. Медленнее всего возрастает величина выигрыша для линейных запросов при построении левоглубоких деревьев поиска. Это объясняется малым числом деревьев для данного случая, поэтому выигрыш от использования memo-структуры начинает ощущаться только для запросов соединения шести 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 для этих случаев, а также тем, что при подсчете учитывается число 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 объеме оперативной памяти, так и в быстродействии оптимизатора запросов СУБД.

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

На основе полученных формул оценена эффективность использования memo-структуры для представления пространства поиска, и сделан вывод, что использование memo-структуры и мультивыражений позволяет получить значительный выигрыш в объеме оперативной памяти и быстродействии оптимизатора СУБД для запросов соединений, включающих более пяти таблиц базы данных.

СПИСОК ЛИТЕРАТУРЫ

1. Ono K., Lohman G. Measuring the Complexity of Join Enumeration in Query Optimization // Proc. of the 16th International Conf. 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 Conf. (Dublin, 1993). – P. 493–504.
3. Hilton P., Pedersen J. Catalan numbers, their generalizations and their uses // The Mathematical Intelligencer. – 1991. – № 13. – P. 64–75.
4. Graefe G., McKenna W. The Volcano Optimizer Generator: Extensibility and Efficient Search // Proc. of the 12th 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. – 1995. – V. 18. – № 3. – P. 19–29.
6. Корн Г., Корн Т. Справочник по математике. – М.: Наука, 1984.

Статья поступила в редакцию 21.03.2003

Юрий Александрович Григорьев родился в 1951 г., окончил в 1975 г. МВТУ им. Н.Э. Баумана. Д-р техн. наук, профессор кафедры “Системы обработки информации и управления" МГТУ им. Н.Э. Баумана. Автор 60 научных работ в области систем обработки информации.

Yu.A. Grigoriev (b. 1951) graduated from the Bauman Moscow Higher Technical School in 1975. Ph.D. (Eng.), professor of “Systems of Data Processing and Control” department of the Bauman Moscow State Technical University. Author of 60 publications in the field of data processing systems.