УДК 004.023

Ю. О. Чернышев, д-р техн. наук, **Н. Н. Венцов**, канд. техн. наук
Государственное образовательное учреждение
высшего профессионального образования
Донской государственный технический университет
(Россия, 344000, Ростов-на-Дону, пл. Гагарина, 1,
тел. 79185991645, e-mail: chernyshevyo@list.ru, myvnn@list.ru)

Генетический алгоритм решения задачи выбора оптимального порядка соединения распределенных отношений

Разработан генетический алгоритм решения задачи выбора оптимального порядка соединения отношений. Описаны процедуры кодирования и декодирования решений, операторы мутации, приведена структурная схема генетического алгоритма.

Розроблено генетичний алгоритм розв'язку задачі вибору оптимального порядку з'єднання відносин. Описано процедури кодування та декодування рішень, операторів мутації та наведено структурну схему генетичного алгоритму.

К л ю ч е в ы е с л о в а: генетический алгоритм, оптимизация, соединение отношений.

Описание проблемы. Имеется множество отношений $R = \{r_0, r_1, ..., r_{n-1}\}$, расположенных на различных узлах сети и подлежащих соединению. Каждое отношение из R находится на уникальном узле сети, т.е. не существует двух отношений, расположенных на одном и том же сервере компьютерной сети. В связи с этим соединению двух отношений предшествует пересылка одного из них на узел, где хранится второе соединяемое отношение. В рассматриваемом случае время передачи данных по сети существенно больше времени их обработки в оперативной памяти. Поэтому затраты на операцию соединения отношений целесообразно оценивать, исходя из объемов передаваемых по сети данных.

Для минимизации сетевого трафика при соединении двух отношений, r_i и r_j, пересылке подлежит наименьшее из них. Затраты на соединение отношений r_i и r_j определим в виде

$$\min(T(r_i), T(r_i)), \qquad (1)$$

где $T(r_i)$ — число кортежей в отношении r_i (мощность отношения r_i); min — функция, возвращающая значение наименьшего из своих аргументов.

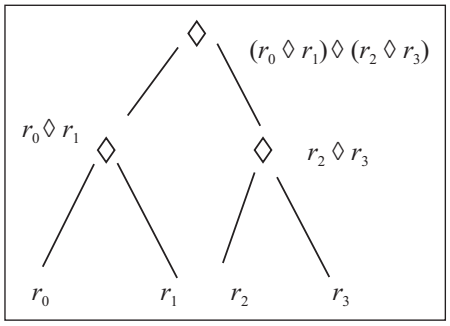


Рис. 1

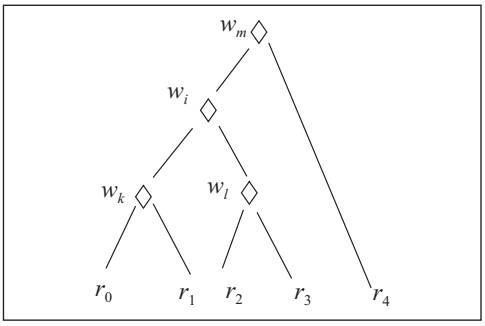


Рис. 2

В случае соединения нескольких отношений порядок соединения можно представить в виде дерева. Изображенное на рис. 1 дерево описывает процесс соединения четырех отношений: $r_0, ..., r_3$. На первом этапе соединяются две пары отношений: r_0 с r_1 и r_2 с r_3. Поскольку пересылке подлежит наименьшее отношение в каждой соединяемой паре, затраты на соединение r_0 с r_1 и r_2 с r_3 определяются соответственно как $\min(T(r_0), T(r_1))$ и $\min(T(r_2), T(r_3))$. Отношение, полученное в результате соединения r_0 с r_1, обозначим $r_0 \lozenge r_1$, а отношение, полученное в результате соединения r_2 с r_3, обозначим $r_2 \lozenge r_3$.

Следующий шаг — соединение отношений $r_0 \lozenge r_1$ и $r_2 \lozenge r_3$. Затраты на операцию соединения отношений определяем по формуле $\min(T(r_0 \lozenge r_1), T(r_2 \lozenge r_3))$. Следовательно, суммарные затраты на соединение отношений в соответствии с порядком, изображенным на рис. 1, определяются так:

$$\min(T(r_0),\ T(r_1)) + \min(T(r_2),\ T(r_3)) + \min(T(r_0 \lozenge r_1),\ T(r_2 \lozenge r_3)).$$

Следует заметить, что в общем случае в зависимости от исходных данных могут быть различные соотношения мощностей исходных, промежуточных и результирующих отношений, например $\min(T(r_0),\ T(r_1)) > \min(T(r_0 \lozenge r_1)$ или $T(r_1) < T(r_0 \lozenge r_1) < T(r_0)$ и т.д.

При подобном подходе к соединению и пересылке отношений результирующее отношение, являющееся результатом соединения всех отношений из R, может оказаться на любом узле сети. Отсутствие привязки результирующего отношения к конкретному узлу сети вполне приемлемо, когда над результирующим отношением необходимо провести вычислительные операции, мощность результата которых априори существенно меньше мощности результирующего отношения.

Формулировка задачи. Обозначим w подмножество внутренних вершин дерева соединения отношений (рис. 2). Каждой внутренней вершине соответствует отношение, полученное в результате объединения отношений, соответствующих двум дочерним вершинам, например для w_m дочерними являются w_i и r_4. Вершине w_i будет соответствовать отношение, полученное в результате соединения отношений $r_0 \lozenge r_1$ и $r_2 \lozenge r_3$, соответствующих вершинам w_k и w_l. При наличии одного атрибута соединения мощность отношения, соответствующего вершине w_i, определяется из выражения [1]

$$T(w_i) = T(w_k \lozenge w_l) = \frac{T(w_k)\,T(w_l)}{\max(V(w_k x),\ V(w_l y))}, \tag{2}$$

где $T(w_i)$ — число кортежей в отношении, соответствующем вершине w_i; $T(w_k \lozenge w_l)$ — число кортежей в отношении, являющемся результатом натурального соединения отношений, соответствующих вершинам w_k и w_l; $V(w_k, x)$ — число различных значений атрибута $x_3 r_3$ отношения, соответствующего вершине w_k; x, y — атрибуты отношений, по которым выполняется соединение. Если имеется два общих атрибута соединения, то

$$T(v_k \lozenge w_l) = \frac{T(w_k)\,T(w_l)}{\max(V(w_k, x1),\ V(w_l, y1))\max(V(w_k x2),\ V(w_l y2))}, \tag{3}$$

где $T(w_k)$ — число кортежей в отношении, соответствующем вершине w_k; $T(w_k \lozenge w_l)$ — число кортежей в отношении, являющемся результатом натурального соединения отношений, соответствующих вершинам w_k и w_l; $V(w_k x)$ — число различных значений атрибута x отношения, соответствующего вершине w_k; $x1$, $x2$, $y1$, $y2$ — атрибуты отношений, по которым выполняется соединение. Если соединяемые отношения не имеют общего атрибута, мощность результирующего отношения определяется как декартово произведение:

$$T(v_k \lozenge w_l) = T(w_k)\,T(w_l). \tag{4}$$

Затраты, связанные с соединением отношений, соответствующих дочерним вершинам w_i, обозначим

$$S(w_i) = \min(T(w_k),\ T(w_l)). \tag{5}$$

Поскольку для получения итогового отношения необходимо осуществить соединение всех вершин, определим величину $S(w_i)$ для каждой внут-

ренней вершины дерева соединения. Тогда критерий оптимальности запишем в виде

$$\sum_{w_i \in W} S(w_i) \to \min . \tag{6}$$

Разработка генетического алгоритма. Оптимальное решение задачи выбора порядка соединения отношений, расположенных на нескольких узлах, может соответствовать произвольному дереву соединения [1]. В связи с этим область поиска решения задачи оптимизации образует два множества: топологических форм деревьев соединения и порядков размещения листьев дерева соединения (r_0 — r_4). Число топологических форм деревьев соединения подчиняется рекуррентному правилу [1]:

$$FD(1) = 1,$$
$$FD(n) = \sum_{i=1}^{n-1} FD(i)\, FD(n-i), \tag{7}$$

где n — число соединяемых отношений. Число различных порядков размещения листьев для заданной формы дерева соединения отношений определяется как число перестановок и равно $n!$

Таким образом, число возможных порядков соединения (NPS) n-отношений в ситуациях, когда не накладываются ограничения на структуру дерева соединения отношений, определяется соотношением [1]

$$NPS(n) = FD(n)\, n!$$

В настоящее время для решения оптимизационных задач, имеющих большую область поиска, эффективно используются генетические алгоритмы, важной особенностью которых является оперирование не с решениями задачи, а с кодами этих решений. Поэтому необходимо разработать процедуры кодирования и декодирования решений. Каждое решение предлагается кодировать особью αR, состоящей из двух хромосом: $\alpha R = (H1, H2)$. Хромосома $H1$ содержит информацию о порядке следования листьев дерева соединения, $H2$ — о структуре дерева (его топологической форме).

Хромосома $H1$ представляет собой кортеж, содержащий идентификаторы соединяемых отношений. Рассмотрим структуру хромосомы $H2$. Введем алфавит $\gamma = \{X, \bullet\}$. По аналогии с [2—5] структуру дерева соединения зададим на базе алфавита $\gamma = \{X, \bullet\}$, используя польскую запись, где X — соответствует соединяемым отношениям (листьям дерева соединения, вершинам подножья), а $\bullet$ — операции соединения (см. рис. 2).

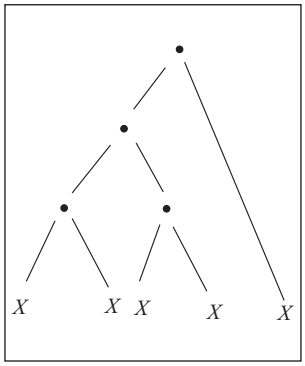


Рис. 3

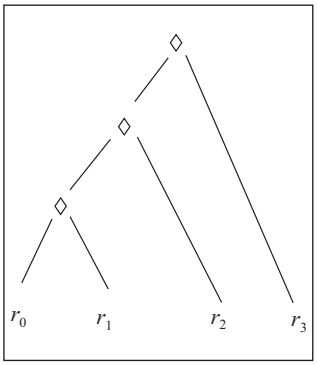


Рис. 4

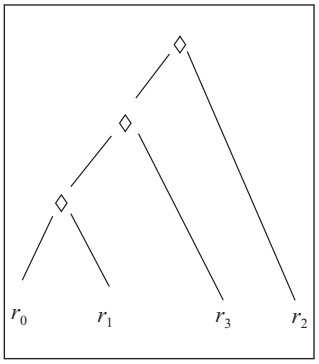


Рис. 5

В рассматриваемом случае символ • соответствует символу ◊. Для восстановления дерева соединения согласно польской записи необходимо просматривать выражение слева направо, символ • соответствует соединению и объединяет два ближайших подграфа, расположенных слева от символа • в польской записи и образованных на предыдущих шагах. Каждая внутренняя вершина, соответствующая оператору •, характеризует отношение, полученное в результате объединения двух подграфов.

Польское выражение для дерева соединения пяти отношений, изображенного на рис. 3, имеет вид $XX \bullet XX \bullet\bullet X \bullet$, для дерева, изображенного на рис. 2, — $r_0 r_1 \bullet r_2 r_3 \bullet\bullet r_4 \bullet$, для дерева, изображенного на рис. 1, — $r_0 r_1 \bullet r_2 r_3 \bullet\bullet$. Обозначим n_X число элементов в польском выражении типа X, где $n_\bullet$ — число символов •. Тогда справедливо равенство $n_x = n_\bullet + 1$ [2—5]. Первый символ • может появиться в польском выражении только после двух символов X. Если в польском выражении провести справа от знака • сечение, то слева от сечения число знаков X больше числа знаков •, по крайней мере, на единицу [2—5].

Если пронумеровать позиции между знаками X в дереве соединения отношений $X\, X_^1\, X_^2\, X_^3 \ldots X_^{n_x - 1}$, то максимальное число знаков в нем может оказаться равным номеру позиции [2]. Если польское выражение соответствует указанным выше условиям, то на его основе можно построить дерево соединения отношений. Число различных форм деревьев соединения определяется по формуле (7).

Таким образом, каждое решение кодируется структурой $\alpha R = (H1, H2)$, состоящей из двух хромосом. Хромосома $H1$ содержит информацию о разметке множества вершин, хромосома $H2$ — о структуре дерева. Поскольку оптимальное решение может соответствовать дереву соединения произвольной формы, при организации процесса генетического поиска

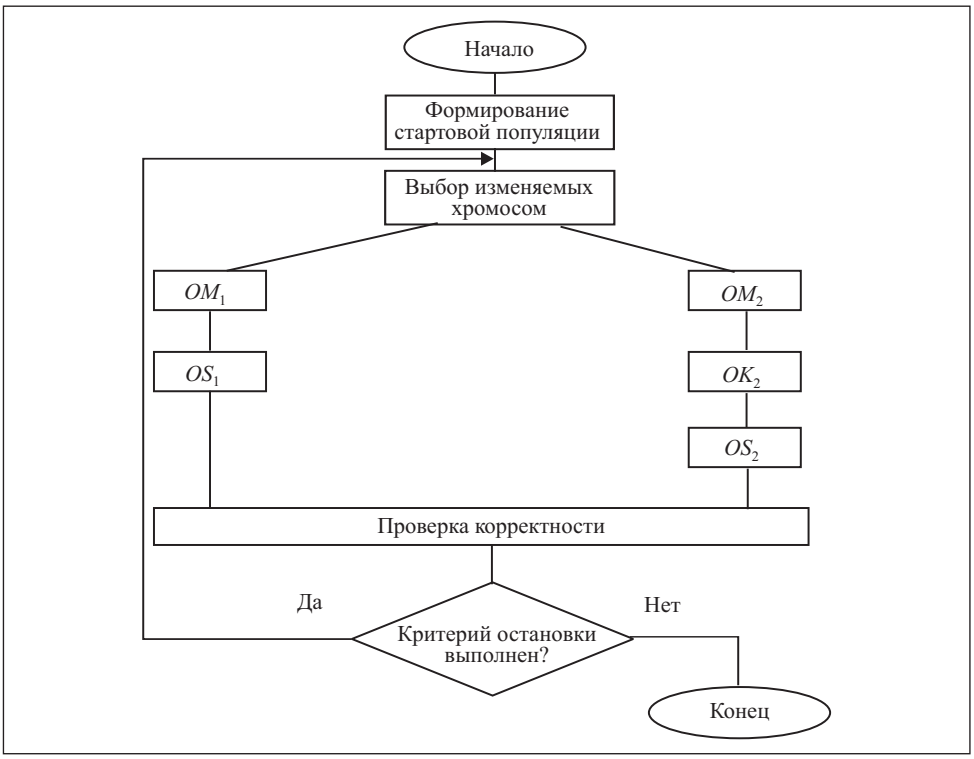


Рис. 6

необходимо применять операторы, модифицирующие не только расположения листьев дерева соединения отношений, но и его структуру.

В качестве примера оператора модификации рассмотрим одноточечный оператор мутации, применяемый к хромосомам особи $\alpha R = (H1, H2)$:

$$H1 = (r_0, r_1, r_2, r_3), \quad H2 = (XX \bullet X \bullet X \bullet).$$

Порядок соединения отношений, кодируемый особью αR, представлен на рис. 4. Затраты на соединение отношений в соответствии с порядком, заданным особью αR, определяются так:

$$\min(T(r_0), T(r_1)) + \min(T(r_0 \lozenge r_1), T(r_2)) + \min(T(r_0 \lozenge r_1 \lozenge r_2), T(r_3)).$$

Применение оператора мутации к хромосоме $H1$ особи αR изменит порядок следования листьев дерева. Например, взаимные перестановки элементов r_2 и r_3 приводят к созданию новой особи:

$$\alpha R' = (H1', H2'), \; H1' = (r_0, r_1, r_3, r_2), \; H2' = H2.$$

Порядок соединения отношений, описываемый особью αR, представлен на рис. 5. Затраты на соединение отношений в соответствии с порядком, заданным особью αR, определяются по формулам (1)—(6):

$$\min(T(r_0), T(r_1)) + \min(T(r_0 \lozenge r_1), T(r_3)) + \min(T(r_0 \lozenge r_1 \lozenge r_3),\ T(r_2)).$$

Мутация хромосомы $H2$ приведет к изменению топологической формы дерева соединения отношений. Например, взаимная перестановка двух элементов, непосредственно предшествующих крайнему правому элементу хромосомы $H2$, приведет к созданию новой особи:

$$\alpha R'' = (H1'', H2''),\ H1'' = H1,\ H2 = (XX \bullet XX \bullet\bullet).$$

Полученный порядок соединения отношений идентичен приведенному на рис. 1.

На основе предложенных процедур кодирования (декодирования) решения, операторов кроссинговера, селекции и мутации разработан генетический алгоритм (рис. 6), особенностью структурной схемы которого является наличие двух групп генетических операторов для $H1$ и $H2$. Генетический алгоритм обладает полиномиальной вычислительной сложностью. В зависимости от настроек (числа особей стартовой популяции, правил формирования новых популяций, критерия остановки и др.) его вычислительная сложность может изменятся в пределах от $O(n^2)$ до $O(n^5)$.

Выводы

Разработанный генетический алгоритм позволяет решать задачу выбора оптимального порядка соединения распределенных отношений за полиномиальное время. Применяемые операторы кодирования (декодирования) решений дают возможность не ограничивать область поиска моделью левостороннего дерева.

Genetic algorithm for solving the problem of optimal choice of the order of relations combination has been designed. The procedures of coding and decoding of solutions, mutation operators have been described, the structural algorithm scheme is presented.

1. *Гарсиа-Молина Г., Ульман Д., Уидом Д.* Системы баз данных. Полный курс.: Пер. с англ. — М. : Изд. дом «Вильямс», 2003. — 1088 с.
2. *Курейчик В. М., Лебедев Б. К., Лебедев О.Б* . Поисковая адаптация: теория и практика. — М. : ФИЗМАТЛИТ, 2006. — 272 с.
3. *Лебедев Б. К.* Адаптация в САПР. —Таганрог: изд-во ТРТУ, 1999. — 160 с.

4. *Лебедев Б. К.* Методы поисковой адаптации в задачах автоматизированного проектирования СБИС: Таганрог: изд-во ТРТУ, 2000. — 192 с.
5. *Курейчик В. М., Лебедев Б. К., Лебедев О. Б., Чернышев Ю. О.* Адаптация на основе самообучения. — Ростов н/Д : изд-во РГАСХМ ГОУ, 2004. — 146 с.

Поступила 04.05.11;
после доработки 16.01.12

ЧЕРНЫШЕВ Юрий Олегович, д-р техн. наук, профессор кафедры «Вычислительные системы и информационная безопасность» Государственного образовательного учреждения высшего профессионального образования Донского государственного технического университета. В 1959 г. окончил Таганрогский радиотехнический ин-т. Область научных исследований — эволюционное моделирование, адаптивные алгоритмы, методы оптимизации.

ВЕНЦОВ Николай Николаевич, канд. техн. наук, доцент кафедры «Информационные технологии» Государственного образовательного учреждения высшего профессионального образования Донского государственного технического университета. В 2003 г. окончил Ростовскую-на-Дону государственную академию сельскохозяйственного машиностроения. Область научных исследований — адаптивная оптимизация.