УДК 681.3

Г.К. КЛАДОВ, канд. физ.-мат. наук, НТУ «ХПИ» (г. Харьков)

РАЗЛОЖЕНИЕ ОТНОШЕНИЯ В СУММУ ПРЯМЫХ ПРОИЗВЕДЕНИЙ

Сформульовані дві теореми о розбитті відношення та таблиці у суму прямих добутків. Це розбиття узагальнює поняття залежності та незалежності стовпців таблиці. Результати можуть бути корисні при архівуванні баз даних.

Are formulated two theorems of decomposition of the relation and table in the sum of direct products. This splitting generalizes concept of dependence and independence columns of the table. The results can be useful at archiving of databases.

Постановка проблемы. Исследуется возможность сократить объем базы данных за счет использования зависимости и независимости столбцов таблиц. Эта задача актуальна для баз данных, которые подготавливаются для длительного хранения.

Анализ литературы. Настоящая статья продолжает исследования, начатые в [1]. При архивации баз данных появляется возможность уменьшить объем данных, используя функциональные связи между столбцами таблицы или полное отсутствие каких-либо связей (независимость).

Нормальные формы [2, 3] таблиц отражают представления разработчика базы данных о функциональной зависимости столбцов таблицы на момент создания базы данных. В этот момент может быть учтена зависимость, которая имеет корни в причинно-следственных связях между сущностями моделируемой области.

Дополнительные возможности возникают при выводе базы данных из активной эксплуатации. В силу различных причин столбцы таблицы могут оказаться (формально) функционально зависимыми, либо независимыми, несмотря на то, что естественных причин для этого нет. Возможность использовать зависимости для уменьшения объема хранимых данных интенсивно изучается [4 – 5], однако только в простейших случаях: функциональной зависимости и полной независимости.

Полная независимость столбцов означает, что таблица может быть представлена как прямое произведение двух более простых таблиц, которые и хранятся. Это позволяет сократить объем базы данных.

В [1] развит формальный метод определения независимости и функциональной зависимости столбцов таблицы на основе показателя «взаимной информативности столбцов». Как показывает практика [2, 6], чистые «зависимость» и «независимость» столбцов таблицы – редкое явление. Как правило, наблюдается нечто среднее – иногда ближе к «функциональной

зависимости», иногда – к «независимости». Изучение этих промежуточных случаев, на наш взгляд, может дать существенный эффект при подготовке баз данных к хранению в хранилищах данных [7 – 8].

Цель статьи. Как оптимально использовать частичную зависимость или независимость для сокращения объема данных? К сожалению, пока полного ответа на этот вопрос нет. Однако первый шаг на этом пути должен состоять в том, чтобы представить произвольную таблицу в виде комбинации прямых произведений других, меньших по объему, таблиц.

Одно из возможных разложений дают теоремы настоящей статьи.

Разложение отношения. Пусть A, B – конечные множества, R – отношение на A, B, то есть подмножество прямого произведения $A*B$. В теории баз данных A и B – домены. Каждое из множеств A, B может, в свою очередь, быть прямым произведением, но это обстоятельство никак не отражается на полученных здесь результатах, поэтому мы считаем, что A и B – произвольные конечные множества.

Пусть $pr_1 = \{a \in A \mid \exists b(a, b) \in R\}$ и $pr_2 = \{b \in B \mid \exists a(a, b) \in R\}$ – проекции отношения R на первую и вторую координаты. Без ограничения общности дальнейших результатов, будем считать, что проекции pr_1, pr_2 совпадают с A и B, соответственно. В противном случае из A и B можно убрать лишние элементы (это возможно для базы данных, предназначенной для длительного хранения).

Обозначим для $a \in A$ через $S(a)$ сечение множества B по элементу a:

$$S(a) = \{b \in B \mid (a,b) \in R\}.$$

Оговоренные выше условия гарантируют, что сечения $\{S(a),\ a \in A\}$, образуют покрытие B.

Очевидно, что $R = A*B$ тогда и только тогда, когда для всех $a \in A$ сечения $S(a)$ совпадают: если R – прямое произведение, то любое сечение $S(a)$ равно B, если все сечения совпадают, то $B = S(a)$.

Для хранения отношения $R = A*B$ как прямого произведения достаточно хранить отдельно A и B, что значительно экономит память.

Обозначим $S^1 = S$ и $S^0 = B \backslash S$. С помощью всех сечений $\{S(a),\ a \in A\}$ создадим разбиение множества B, а именно, для каждого $U \subseteq A$ положим

$$Q(U) = \cap (S(a)^{\sigma(a)},\ a \in A, \text{ где } \sigma(a) = 1 \text{ при } a \in U,\ \sigma(a) = 0 \text{ при } a \notin U.$$

Лемма. Непустые множества из $\{Q(U),\ U \subseteq A\}$, образуют разбиение множества B.

Доказательство. Множества $Q(U)$ обладают следующими свойствами:

1) каждый элемент $u \in U$ входит в R вместе с каждым $q \in Q(U)$. Действительно, u входит в R вместе с каждым элементом $S(u)$, а $Q(U) \subseteq S(u)$.

2) если $U \neq V$, то $Q(U) \cap Q(V) = \varnothing$. Если $u \in U$, $u \notin V$, то в пересечение $Q(U) \cap Q(V)$ входит $S(u)^1 \cap S(u)^0 = \varnothing$.

3) $B = \cup Q(U)$ по всем $U \subseteq A$. Пусть $b \in B$. Соберем в множество U все $u \in A$, которые входят в R вместе с b. Тогда $b \in Q(U)$. Действительно, $Q(U) = \cap (S(a)^{\sigma(a)}$, $\sigma(a) = 1$ при $a \in U$, $\sigma(a) = 0$ при $a \notin U$). Для $a \in U$ сечение $S(a)$ содержит b. Для $a \notin U$ сечение $S(a)$ не содержит b по построению U. Поэтому $S(a)^0$ содержит b. Следовательно, все множества в пересечении содержат b. Тем самым доказано, что существует множество U, такое, что $b \in Q(U)$. Убрав пустые множества из набора $\{Q(U), U \subseteq A\}$, получаем разбиение B.

Теорема 1. Множества $\{U*Q(U), U \subseteq A\}$, образуют разбиение множества R:

$$R = \cup U * Q(U), U \subseteq A.$$

Доказательство. Каждый элемент из U входит в R с каждым элементом из $Q(U)$. Остальное следует из определений и леммы.

Разложение таблицы. Если отношение – подмножество, то таблица – подмножество с повторениями. Следуя [1], будем рассматривать таблицу как функцию

$$\tau : A*B \text{ -> } N.$$

Здесь и далее N – дополненное нулем множество натуральных чисел.

Таблицу можно трактовать и как матрицу T с натуральными элементами, у которой множество индексов строк равно A, множество индексов столбцов равно B.

По [1] столбцы A, B таблицы T независимы, если существуют функции $\tau_1(a)$, $\tau_2(b)$, такие, что для всех $(a, b) \in A*B$ имеем $\tau(a,b) = \tau_1(a)*\tau_2(b)$. На языке матриц будем говорить, что индексы матрицы T независимы, если существуют вектор-столбец $t = (\tau_1(a), a \in A)$ и вектор-строка $s = (\tau_2(b), b \in B)$, такие, что $T = t*s$.

Матрицу $M : U * V \text{ -> } N$, $U \subseteq A$, $V \subseteq B$, доопределим нулями на парах из $(A * B) \setminus (U * V)$. Ее сумма с матрицей T:

$(T+M)(a, b) = T(a, b) + M(a, b)$, если $a \in U$ and $b \in V$,
$(T+M)(a, b) = T(a, b)$, если $a \notin U$ or $b \notin V$.

Идею разложения проиллюстрируем на матрице с одним столбцом T. В этом случае нам нужно представить матрицу в виде суммы столбцов с одинаковыми коэффициентами, отличными от нуля.

Положим $T_0 = T$, $A_0 = A$, $i = 0$.

Применим следующий алгоритм для разложения в сумму столбцов:

1) если в столбце T_0 есть нулевой элемент, удалим соответствующий индекс из A_0, получим A_1. Если множество A_1 пустое, то закончим работу, иначе удалим соответствующую строку из T_0, получим T_1. Переобозначим A_1 и T_1 как A_0 и T_0, положим $i = i + 1$ и перейдем к следующему пункту;

2) найдем минимальный элемент в столбце T_0, пусть он будет b. Создадим вектор $t_i = A_0*\{b\}$. Положим $T_0(a) = T_0(a) - t_i$. Вернемся к п. 1.

Векторы $t_1, t_2, \ldots$ представляют собой требуемое разложение.

Алгоритм заканчивает свою работу в конечное число шагов, так как каждый раз число элементов в множестве A_0 уменьшается.

Пример работы алгоритма (он делает два шага):

$$\begin{pmatrix}5\\9\\2\end{pmatrix} = \begin{pmatrix}2\\2\\2\end{pmatrix} + \begin{pmatrix}3\\7\\0\end{pmatrix} = \begin{pmatrix}2\\2\\2\end{pmatrix} + \begin{pmatrix}3\\3\\0\end{pmatrix} + \begin{pmatrix}0\\4\\0\end{pmatrix}$$

Вернемся к матрицам. Введем некоторые дополнительные обозначения.

а) Обозначим $\div$ неотрицательную разность: $x \div y = 0$, если $x < y$, и $x - y$, если $x \geq y$.

б) Пусть $U \subseteq A$. Для $b \in B$ определим число $\tau(U, b)$ как

$$\tau(U, b) = min(T(a, b), a \in U) \div max(T(a, b), a \notin U).$$

Обозначим $U*\tau(U, b)$ вектор-столбец с компонентами $\tau(U, b)$ для строк из U и нулями во всех остальных строках из A. Если $(p, q, \ldots, z)$ – вектор-строка, то $U*(p, q, \ldots, z)$ – матрица, у которой строки, индексированные U, равны $(p, q, \ldots, z)$, а остальные равны $(0, 0, \ldots, 0)$.

Теорема 2. $\boldsymbol{T = \Sigma(U*(\tau(U, b), b \in B), U \subseteq A)}$.

Доказательство. Утверждение достаточно доказать для каждого столбца в отдельности.

Рассмотрим сначала случай, когда в столбце b все элементы матрицы T различны. Перенумеруем элементы A по возрастанию $T(a, b)$, так что $T(a_1, b) < T(a_2, b) < \ldots < T(a_m, b)$. Обозначим $U_0 = A$, $U_1 = A\backslash\{a_1\}$, $U_2 = A\backslash\{a_1, a_2\}$, $\ldots$, $U_{m-1} = \{a_m\}$. Покажем, что для любого U, отличного от U_i, $\tau(U, b) = 0$. Действительно, поскольку U не совпадает ни с одним U_i, то для некоторых s и p, $s < p$, имеем $a_s \in U$ и $a_p \notin U$. Выберем минимальное s и максимальное p из всех индексов, удовлетворяющих этим условиям. Тогда

$$\tau(U, b) = T(a_s, b) \div T(a_p, b) = 0.$$

Для множеств U_i непосредственные вычисления дают значения

$$\tau(U_0, b) = T(a_1, b), \ldots, \tau(U_i, b) = T(a_{i+1}, b) - T(a_i, b), \ldots, \tau(U_{m-1}, b) = T(a_m, b) - T(a_{m-1}, b).$$

Прямое произведение $U_i*\tau(U_i, b)$ – это матрица-столбец, в строке a которой содержится $\tau(U_i, b)$, если $a \in U_i$, и 0, если $a \notin U_i$.

Найдем сумму коэффициентов для a_k по всем U_i, содержащим a_k:

$$\Sigma(\tau(U_i, b), a_k \in U_i) = \Sigma(\tau(U_i, b), U_i = U_0, \ldots, U_{k-1}) = T(a_1, b) + (T(a_2, b) - T(a_1, b)) + \ldots + (T(a_{k-1}, b) - T(a_{k-2}, b)) + (T(a_k, b) - T(a_{k-1}, b)) = T(a_k, b).$$

Таким образом, левая и правая части в утверждении совпадают для любого столбца.

Случай, когда не все элементы столбца различны, сводится к предыдущему отождествлением строк. Множества U_i уменьшаются тогда каждый раз на множество строк, имеющих в столбце одинаковые значения.

Другое разложение таблицы можно сделать, поменяв ролями множества A и B: $T = \Sigma((V*\tau(V, a), a \in A), V \subseteq B)$.

Эти разложения могут иметь различное количество слагаемых, как показывает следующий пример:

16	70	92
90	92	199

=

16	70	92
16	70	92

+

0	0	0
74	70	107

16	70	92
90	92	199

=

0	0	22
0	0	107

+

0	54	54
0	2	2

+

16	16	16
90	90	90

Выводы. Предложен алгоритм разложения таблицы в сумму прямых произведений. В случае, когда таблица разлагается в небольшое количество компонент, разложение дает возможность сэкономить память, отводимую под таблицу. Разложение в прямую сумму не единственно, поэтому нужен поиск оптимального разложения по критерию экономии памяти. Это будет задачей дальнейших исследований.

Список литературы: **1.** *Кладов Г.К.* Свойства таблиц как носителей данных // Вестник НТУ «ХПИ». – 2002. – Т.7. – № 9. – С. 91–96. **2.** *Коннолли Т., Бегг К., Страчан А.* Базы данных: проектирование, реализация, сопровождение. Теория и практика. 2-е изд.: Уч. пос. – М.: Издательский дом «Вильямс», 2000. – 1120 с. **3.** *Дейт К. Дж.* Введение в системы баз данных. 7-е изд.: Пер. с англ. – М.: Издательский дом «Вильямс», 2002. – 1072 с. **4.** *Fagin R.* Normal form and Relational Database Operators // Proc. 1979 ACM SIGMOD Intern. Conf. on Management of Data. – Boston, Mass., 1979. **5.** *Nicolas J.M.* Mutual Dependencies and Some Results on Undecomposable Relations // Proc. 4th Intern. Conf. on Very Large Data Bases. – Berlin, FDR., 1978. **6.** *Чекалов А.П.* Базы данных: от проектирования до разработки приложений. – СПб.: БХВ-Петербург, 2003. – 384 с. **7.** *Inmon W.H.* Building the Data Warehouse. – New York, NY: John Willey & Sons. – 1993. **8.** *Devlin B.* Data Warehouse: From Architecture to Implementation. Harlow: Addison Wesley Longman. – 1997.

Поступила в редакцию 12.04.2005