Krassimir Markov, Vitalii Velychko, Oleksy Voloshin
(editors)

Natural and Artificial Intelligence

ITHEA
SOFIA
2010

Krassimir Markov, Vitalii Velychko, Oleksy Voloshin (ed.)

Natural and Artificial Intelligence

ITHEA®

Sofia, Bulgaria, 2010

ISBN 978-954-16-0043-9

First edition

Recommended for publication by The Scientific Concil of the Institute of Information Theories and Applications FOI ITHEA

This book is engraved in prof. Zinoviy Lvovich Rabinovich memory. He was a great Ukrainian scientist, co-founder of ITHEA International Scientific Society (ITHEA ISS). To do homage to the remarkable world-known scientific leader and teacher this book is published in Russian language and is concerned to some of the main areas of interest of Prof. Rabinovich.

The book is opened by the last paper of Prof. Rabinovich specially written for ITHEA ISS. Further the book maintains articles on actual problems of natural and artificial intelligence, information interaction and corresponded intelligent technologies, expert systems, robotics, classification, business intelligence; etc. In more details, the papers are concerned in:conceptual problems of the natural and artificial intelligent systems: structures and functions of the human memory, ontological models of knowledge representation, knowledge extraction from the natural language texts;network technologies;evolution and perspectives of development of the mechatronics and robotics;visual communication by gestures and movements, psihology of vision and information technologies of computer vision, image processing;object classification using qualitative characteristics;methods for comparing of alternatives and their ranging in the procedures of expert knowledge processing;ecology of programming – a new trend in the software engineering;decision support systems for economics and banking;systems for automated support of disaster risk management;and etc.

It is represented that book articles will be interesting for experts in the field of information technologies as well as for practical users.

General Sponsor: Consortium FOI Bulgaria (www.foibg.com).

Printed in Bulgaria

ISBN 978-954-16-0043-9

C\o Jusautor, Sofia, 2010

ОБОБЩЕННАЯ ТАБЛИЧНАЯ АЛГЕБРА, ОБОБЩЕННОЕ ИСЧИСЛЕНИЕ СТРОК И ИХ ЭКВИВАЛЕНТНОСТЬ

Дмитрий Буй, Ирина Глушко

Аннотация: *В данной статье проведено обобщение табличной алгебры, построенной на основе реляционных алгебр Кодда;обобщение состоит в том, что вместо таблиц рассматриваются пары, состоящие из таблиц и схем таблиц. Представлено обобщение классического результата об эквивалентности реляционной алгебры Кодда и исчисление строк (кортежей). Исчисление строк пополнено произвольными предикатными и функциональными сигнатурами на универсальном домене. Доказано, что при этом обобщении исчисление строк остается не менее выразительным, чем табличная алгебра.*

Ключевые слова: *реляционные (табличние) базы данных, исчисление строк, табличная алгебра.*

ACM классификация ключевых слов:*H.2 Database Management (E.5) H.2.4 Systems – Relational databases*

Введение

Уточнение реляции в терминах именных множеств осуществлено Редьком В.Н., Броной Ю,И., Буем Д.Б., Поляковим С.А. в монографии [Редько, Брона, Буй, Поляков, 2001]. В данной работе обобщено полученные результаты на таблицы, которым приписывается определенная схема. Часть статьи посвящена обобщению классического результата об эквивалентности реляционной алгебры Кодда и исчисления строк (кортежей), пополненного произвольными предикатными и функциональными сигнатурами на универсальном домене. Доказано, что при этом исчисление строк остается не менее выразительным, чем табличная алгебра.

Обобщение табличной алгебры

Все неопределенные здесь понятия и обозначение понимаем в смысле [Редько, Брона, Буй, Поляков, 2001]. Рассматриваем два множества: $\boldsymbol{A}$ – множество атрибутов и $\boldsymbol{D}$ – универсальный домен. Под табличной алгеброй понимаем алгебру $\langle \boldsymbol{T}, \Omega_{P,\Xi} \rangle$, где $\boldsymbol{T}$ – множество всех таблиц, $\Omega_{P,\Xi}$ = $\{\cup_{\boldsymbol{R}}, \cap_R, \setminus_R, \sigma_{p,R}, \pi_{X,R}, \underset{R_1,R_2}{\otimes}, \div^{R_1}_{R_2}, Rt_{\xi,R}, \sim_R\}^{p\in P,\xi\in\Xi}_{X,R,R_1,R_2\subseteq A}$ – сигнатура, P, Ξ – множества параметров. Произвольное конечное множество атрибутов $R \subseteq \boldsymbol{A}$ назовём схемой. Строкой схемы R называется именное множество на паре R, $\boldsymbol{D}$, проекция которого по первой компоненте равна R (т.е. рассматривается функция вида $s: R \to D$). Под (обобщенной) таблицей понимаем пару $\langle t, R \rangle$, где $t \in T(R)$ – таблица фиксированной схемы R (в смысле [Редько, Брона, Буй, Поляков, 2001]). Тогда $\boldsymbol{T}(R) = \{\langle t, R \rangle \mid t \in T(R)\}$ – множество всех таблиц схемы R, а $\boldsymbol{T} = \cup \boldsymbol{T}(R)$ – множество всех таблиц. Отличие от определения таблицы в смысле [Редько, Брона, Буй, Поляков, 2001] заключается в том, что каждой таблице приписывается ее схема. По сути это влияет только на случай пустой таблицы $t_\varnothing$, поскольку по непустой таблице схема восстанавливается однозначно. Запись $\langle t_\varnothing, R \rangle$ обозначает пустую таблицу схемы R.

Выражением табличной алгебры называется любое выражение, построенное из таблиц множества $\boldsymbol{T}$ при использовании операций множества $\Omega_{P,\Xi}$.

Зададим операции. Определим сначала теоретико-множественные операции. Под объединением $\cup_R$ (пересечением $\cap_R$, разностью $\setminus_R$) таблиц схемы R понимается бинарная частичная операция, полученная ограничением теоретико-множественного объединения (соответственно пересечения, разности) на множество всех таблиц схемы R.

$\cup_R : \boldsymbol{T} \times \boldsymbol{T} \tilde{\to} \boldsymbol{T}$, $dom \cup_R = \boldsymbol{T}(R) \times \boldsymbol{T}(R)$, $\langle t_1, R \rangle \cup_R \langle t_2, R \rangle = \langle t_1 \cup t_2, R \rangle$;

$\cap_R : \boldsymbol{T} \times \boldsymbol{T} \tilde{\to} \boldsymbol{T}$, $dom \cap_R = \boldsymbol{T}(R) \times \boldsymbol{T}(R)$, $\langle t_1, R \rangle \cap_R \langle t_2, R \rangle = \langle t_1 \cap t_2, R \rangle$;

$\setminus_R : \boldsymbol{T} \times \boldsymbol{T} \tilde{\to} \boldsymbol{T}$, $dom \setminus_R = \boldsymbol{T}(R) \times \boldsymbol{T}(R)$, $\langle t_1, R \rangle \setminus_R \langle t_2, R \rangle = \langle t_1 \setminus t_2, R \rangle$; где $t_1, t_2 \in T(R)$.

Пусть $p : S \tilde{\to} \{true, false\}$ – частичный предикат на множестве строк. Под селекцией по предикату p таблиц схемы R понимается унарная частичная параметрическая операция $\sigma_{p,R}$, которая таблице сопоставляет её подтаблицу, содержащую строки, на которых предикат p истинный. Следовательно, $\sigma_{p,R} : \boldsymbol{T} \tilde{\to} \boldsymbol{T}$, $dom \sigma_{p,R} = \{\langle t, R \rangle \mid t \subseteq dom\ p\}$, $\sigma_{p,R}(\langle t, R \rangle) = \langle \{s \mid s \in t \wedge p(s) \simeq true\}, R \rangle$, где $t \in T(R)$. Здесь и далее $\simeq$ – обобщенное (по другой терминологии сильное) равенство, то есть, обе части или одновременно не определенны или одновременно определенны и равны.

Пусть $X \subseteq \boldsymbol{A}$ – конечное множество атрибутов. Под проекцией по множеству атрибутов X таблиц схемы R понимается унарная параметрическая операция $\pi_{X,R}$, значениями которой есть таблицы схемы $R \cap X$, состоящие из ограничений по множеству атрибутов X строк исходных таблиц. Следовательно, $\pi_{X,R} : \boldsymbol{T}(R) \to \boldsymbol{T}(R \cap X)$, $\pi_{X,R}(\langle t, R \rangle) = \langle \{s \mid X \mid s \in t\}, R \cap X \rangle$, где $t \in T(R)$.

Под соединением таблиц схем R_1, R_2 понимается бинарная операция $\underset{R_1,R_2}{\otimes}$, значениями которой являются таблицы схемы $R_1 \cup R_2$, состоящие из всех объединений совместных строк исходных таблиц.

Следовательно, $\underset{R_1,R_2}{\otimes} : \boldsymbol{T}(R_1) \times \boldsymbol{T}(R_2) \to \boldsymbol{T}(R_1 \cup R_2)$,

$\langle t_1, R_1 \rangle \underset{R_1,R_2}{\otimes} \langle t_2, R_2 \rangle = \langle \{s_1 \cup s_2 \mid s_1 \in t_1 \wedge s_2 \in t_2 \wedge s_1 \approx s_2\}, R_1 \cup R_2 \rangle$, где $t_1 \in T(R_1), t_2 \in T(R_2)$.

Под делением таблиц схемы R_1 на таблицы схемы R_2, где $R_2 \subseteq R_1$, понимается бинарная параметрическая частичная операция $\div^{R_1}_{R_2}$, значениями которой являются таблицы схемы $R_1 \setminus R_2$, состоящие из определенных строк, полученных в результате проекции по множеству атрибутов $R_1 \setminus R_2$ таблиц схемы R_1 (см. ниже). Следовательно, $\div^{R_1}_{R_2} : \boldsymbol{T} \times \boldsymbol{T} \tilde{\to} \boldsymbol{T}$, $dom \div^{R_1}_{R_2} = \boldsymbol{T}(R_1) \times \boldsymbol{T}(R_2)$,

$\langle t_1, R_1 \rangle \div^{R_1}_{R_2} \langle t_2, R_2 \rangle = \left\langle \{s \in \pi_{R_1 \setminus R_2}(t_1) \mid \{s\} \underset{R_1,R_2}{\otimes} t_2 \subseteq t_1\}, R_1 \setminus R_2 \right\rangle$, где $t_1 \in T(R_1)$, $t_2 \in T(R_2)$.

Под активным дополнением таблиц схемы R понимается унарная операция $\sim_R$, которая таблице сопоставляет дополнение в её насыщении. Следовательно, $\sim_R : \boldsymbol{T}(R) \to \boldsymbol{T}(R)$, $\sim \langle t, R \rangle = \langle C(t) \setminus t, R \rangle$, где $t \in T(R)$. Для замкнутости текста отметим, что $C(t) = \underset{A \in R}{\otimes} \pi_{\{A\}}(t)$.

Под переименованием таблиц схемы R, соответствующим инъективной частичной функции переименования атрибутов $\xi : \boldsymbol{A} \tilde{\to} \boldsymbol{A}$, понимается унарная параметрическая операция $Rt_{\xi,R}$, $dom\, Rt_{\xi,R} = \{\langle t,R\rangle | t \in T_{\xi}(R)\}$, где $T_{\xi}(R)$ – множество таблиц схемы R, а схема R ξ-допустима, значения которой задаются равенством $Rt_{\xi,R}(\langle t,R\rangle) = \langle Rs_{\eta,R}[t], \eta[R]\rangle$, $t \in T_{\xi}(R)$, где $\eta = \xi \cup \mathrm{id}_{\boldsymbol{A}\setminus dom\xi}$, $Rs_{\eta,R}(s) = \{\langle \eta(A), s(A)\rangle | A \in R\}$, $s \in S(R)$.

Обобщенное исчисление строк

В основе большинства реляционных языков запросов лежит реляционное исчисление, поскольку в отличие от реляционной алгебры, исчисление выражает лишь то, каким должен быть результат и не предусматривает определение того, как его получить. Реляционное исчисление основывается на исчислении предикатов первого порядка. Есть две формы реляционного исчисления: исчисление с переменными строками (по другой терминологии кортежами) и исчисление с переменными на доменах. Эти формы предложены Е. Коддом (E. Codd) [Codd, 1972] и М. Лакруа (M. Lacroix) с А. Пиротте (A. Pirotte) [Lacroix, Pirotte 1977] соответственно. Данный вопрос также рассматривали в своих роботах Д. Мейер (D. Maier) [Мейер, 1987], Дж. Д. Ульман (J. D. Ullman) [Ульман, 1983], К. Дж. Дейт (C. J. Date) [Дейт, 2005], Т.М. Коннолли (T. M. Connolly) и К. Бегг (C. E. Begg) [Коннолли, Бегг, 2003], В. В. Пасичник и В. А. Резниченко [Пасичник, Резниченко, 1977].

Выражения исчисления строк имеют вид $\{x(R) \,|\, P(x)\}$, где P – некоторый предикат над переменной строкой x, а R – схема. Это выражение обозначает таблицу $\langle t,R\rangle$, $t \in T(R)$, состоящую из всех строк, на которых предикат P истиннен.

Введем множество так называемых разрешенных формул исчисления строк при использовании:

- множества атрибутов $\boldsymbol{A}$ и универсального домена $\boldsymbol{D}$;
- множества предметных переменных (переменных строк) $x_1, x_2, \ldots$;
- множества предметных констант $d_1, d_2, \ldots$;
- множества функциональных символов $f_1^{n_1}, f_2^{n_2}, \ldots$;
- множества предикатных символов $p_1^{m_1}, p_2^{m_2}, \ldots$.

Областью интерпретации предметных констант есть универсальный домен $\boldsymbol{D}$, предметных переменных – множество всех строк. Применяем $\boldsymbol{x}$ как синтаксическую переменную, областью изменения которой являются переменные; $\boldsymbol{f}$ ($\boldsymbol{p}$)как синтаксическую переменную, областью изменения которой являются функциональные (соответственно предикатные) символы; $\boldsymbol{d}$ как синтаксическую переменную, областью изменения которой являются константы; наконец, $\mathcal{A}$ как синтаксическую переменную, областью изменения которой являются атрибуты.

Следующие выражения являются термами (индукция по длине термов):

a) всякая предметная константа есть терм;

b) $\boldsymbol{x}(\mathcal{A})$ – терм;

c) если $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n$ – термы, $\boldsymbol{f}$ – n-арный функциональный символ, то $\boldsymbol{f}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n)$ – терм;

d) выражение является термом только в том случае, если это следует из правил a), b), c).

Будем применять $\boldsymbol{u}$ как синтаксическую переменную, областью изменения которой являются термы. Сформулируем правила построения формул. Атомарные формулы (атомы) бывают двух типов.

a1. Пусть $\langle t,R\rangle$ – таблица, а $\boldsymbol{x}$ – переменная строка. Тогда $t(\boldsymbol{x})$ – атом, означающий, что $\boldsymbol{x} \in t$.

a2. Пусть $\boldsymbol{u}_1,\dots,\boldsymbol{u}_m$ – термы, а $\boldsymbol{p}$ – m-арный предикат на универсальном домене $\boldsymbol{D}$. Тогда $\boldsymbol{p}(\boldsymbol{u}_1,\dots,\boldsymbol{u}_m)$ – атом.

Используем логические связки $\neg$, $\wedge$, $\vee$, кванторы $\exists, \forall$ и скобки $($, $)$ для построения формул из атомов.

f1. Каждый атом – формула

f2. Если P – формула, то $\neg P$ – формула.

f3. Если P и Q – формулы, то $P \wedge Q$, $P \vee Q$ – формулы.

f4. Пусть $\boldsymbol{x}$ – переменная строка, P – формула, $R \subseteq \boldsymbol{A}$ – схема, тогда $\exists \boldsymbol{x}(R)P$ – формула.

f5. Пусть $\boldsymbol{x}$ – переменная строка, P – формула, $R \subseteq \boldsymbol{A}$ – схема, тогда $\forall \boldsymbol{x}(R)P$ – формула.

f6. Если P – формула, то (P) – формула.

f7. Других формул нет.

В общем случае переменные строки в формулах могут быть свободные или связанные. Смысл этих понятий такой же как и в исчислении предикатов: вхождение переменной x в данную формулу называется связанным, если x является переменной входящего в эту формулу квантора ($\exists$ или $\forall$) или находится в области действия входящего в эту формулу квантора; в противном случае вхождение переменной x в данную формулу называется свободным. Переменная называется свободной (связанной) переменной в данной формуле, если существуют свободные (соответственно связанные) ее вхождения в эту формулу [Мендельсон, 1971].

Для каждой переменной строки x определим схему (конечное множество атрибутов) $scheme(x,P)$ и множество атрибутов $attr(x,P)$, с которыми строка x встречается в формулах. Выражения $scheme(x,P)$ и $attr(x,P)$ определены, если строка x имеет свободное вхождение в формулу P, причем имеет место включение $attr(x,P) \subseteq scheme(x,P)$ (что следует из последующих определений при условии определения выражений).

Выделим класс разрешенных формул, используя понятия свободных и связанных переменных строк, схемы и множества атрибутов, с которыми переменная строка встречается в формулах. Определим выражение $attr$ сначала для термов:

1. если $u = \boldsymbol{d}$, то, $attr(\boldsymbol{x},u) = \varnothing$;

2. если $u = \boldsymbol{x}$ ($\boldsymbol{A}$), то $attr(\boldsymbol{x},u) = \{A\}$, а $attr(\boldsymbol{x},\boldsymbol{y}(A)) = \varnothing$, где $\boldsymbol{x} \neq \boldsymbol{y}$;

3. если $\boldsymbol{u} = \boldsymbol{f}(\boldsymbol{u}_1,\dots,\boldsymbol{u}_n)$, где $\boldsymbol{u}_i$ – термы, то $attr(\boldsymbol{x},u) = \bigcup_{i=1}^{n} attr(\boldsymbol{x},\boldsymbol{u}_i)$.

Пусть формула P – атом, тогда

a1. если $P = t(\boldsymbol{x})$, то (единственное) вхождение переменной строки $\boldsymbol{x}$ свободно в формуле P и $scheme(\boldsymbol{x},P) = attr(\boldsymbol{x},P) = R$, где R – схема таблицы t;

a2. если $P = \boldsymbol{p}(\boldsymbol{u}_1,...,\boldsymbol{u}_m)$, где $\boldsymbol{u}_i$ – термы, причем $\boldsymbol{x}_1,...,\boldsymbol{x}_k$ – все переменные этих термов, то вхождения этих переменных строк свободны в формуле P, схема $scheme(\boldsymbol{x}_i,P)$ не определена, а $attr(\boldsymbol{x}_i,P) = \bigcup\limits_{j=1}^{m} attr(\boldsymbol{x}_i,\boldsymbol{u}_j)$, $i = 1,...,k$.

Атомарные формулы всегда разрешены. Для построения всех разрешенных формул проведем индукцию по длине формул. Предположим, что G и Q разрешенные формулы.

f2. Если $P = \neg G$, то формула P – разрешена, а вхождения переменных в P свободны или связаны в зависимости от того, свободны или связаны вхождения этих же переменных в формулу G. Если $\boldsymbol{x}$ входит в формулу G свободно, то $scheme(\boldsymbol{x},P) \simeq scheme(\boldsymbol{x},G)$ и $attr(\boldsymbol{x},P) = attr(\boldsymbol{x},G)$.

f3. Если $P = G \wedge Q$ или $P = G \vee Q$, то вхождения переменных в формулу P свободны или связаны в зависимости от того, свободны или связаны вхождения этих переменных в G или Q. Пусть переменная строка x входит в подформулы G и/или Q свободно. Определим схему и множество атрибутов, с которыми строка $\boldsymbol{x}$ встречается в формулах, для формулы P. Имеют место следующие случаи.

a. Схемы формул $scheme(\boldsymbol{x},G)$ и $scheme(\boldsymbol{x},Q)$ определены. Для того чтобы формула P была разрешена, должно выполнять равенство $scheme(\boldsymbol{x},G) = scheme(\boldsymbol{x},Q)$. Полагаем по определению $scheme(\boldsymbol{x},P) = scheme(\boldsymbol{x},G)$.

b. Схема определена только для одной из подформул. Пусть схема $scheme(\boldsymbol{x},G)$ определена, а схема $scheme(\boldsymbol{x},Q)$ – не определена. Для того чтобы формула P была разрешена, должно быть выполнено включение $attr(\boldsymbol{x},Q) \subseteq scheme(\boldsymbol{x},G)$. Полагаем по определению $scheme(\boldsymbol{x},P) = scheme(\boldsymbol{x},G)$.

c. Схема не определена для обоих подформул. В этом случае и схема $scheme(\boldsymbol{x},P)$ не определена.

В любом случае $attr(\boldsymbol{x},P) = attr(\boldsymbol{x},G) \cup attr(\boldsymbol{x},Q)$.

f4. Если $P = \exists \boldsymbol{x}(R)G$ и переменная $\boldsymbol{x}$ входит в формулу G свободно, то формула P – разрешена. Кроме того, если $scheme(\boldsymbol{x},G)$ определена, то должно выполняться равенство $scheme(\boldsymbol{x},G) = R$ при условии выполнения включения $attr(\boldsymbol{x},G) \subseteq R$. Так как переменная $\boldsymbol{x}$ не входит свободно в формулу P, то $scheme(\boldsymbol{x},P)$ и $attr(\boldsymbol{x},P)$ не определены. Если $\boldsymbol{y} \neq \boldsymbol{x}$, то любое вхождение переменной $\boldsymbol{y}$ в P свободно или связано, в соответствии с тем, свободно или связано вхождение $\boldsymbol{y}$ в G. Если $\boldsymbol{y}$ входит в P свободно, то $scheme(\boldsymbol{y},P) \simeq scheme(\boldsymbol{y},G)$ и $attr(\boldsymbol{y},P) = attr(\boldsymbol{y},G)$.

f5. Если $P = \forall \boldsymbol{x}(R)G$, то все определения и ограничения такие же, как и в случае f4 для квантора существования.

f6. Если $P = (G)$, то P – разрешена, а свободные и связанные вхождение переменных, схема и множество атрибутов, с которыми переменная строка встречается в формулах, остаются теми же, что и у формулы G.

После введения множества разрешенных формул можем дать окончательное определение выражения исчисления строк. А именно, выражение исчисления строк имеет вид $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x})\}$, где

1. формула P – разрешена;
2. переменная $\boldsymbol{x}$ – единственная свободно входящая в формулу P переменная;
3. если $scheme(\boldsymbol{x},P)$ определена, то $scheme(\boldsymbol{x},P)=R$, иначе $attr(\boldsymbol{x},P)\subseteq R$.

Пусть $P(\boldsymbol{x})$ – разрешенная формула, $R\subseteq \boldsymbol{A}$. В результате подстановки конкретной строки s схемы R вместо переменной $\boldsymbol{x}$ в формулу P получим формулу обозначенную $P(s/\boldsymbol{x})$. Определим сначала истинностные значения атомов:

a1. пусть строка $\boldsymbol{x}$ в подформуле $t(\boldsymbol{x})$ свободна в P. Атом $t(\boldsymbol{x})$ приобретает значение истина при подстановке конкретной строки s вместо $\boldsymbol{x}$, если $s\in t$, иначе атом $t(\boldsymbol{x})$ приобретает значение ложь;

a2. пусть строка $\boldsymbol{x}$ в подформуле $\boldsymbol{p}(\boldsymbol{u}_1,...,\boldsymbol{u}_m)$ свободна в P, тогда при подстановке конкретной строки s_i вместо $\boldsymbol{x}$, заменим $\boldsymbol{x}(A_i)$ на $d_i\in \boldsymbol{D}$, где $\langle A_i,d_i\rangle\in s_i$ (т.е. d_i – значение атрибута A_i в строке s_i). Атом $\boldsymbol{p}(\boldsymbol{u}_1,...,\boldsymbol{u}_m)$, где $\boldsymbol{u}_i$ – термы (предметные константы), приобретает значение истина, при подстановке конкретной строки, если предикат $\boldsymbol{p}$ истинный на соответственных значениях, иначе атом приобретает значение ложь.

Набор значений истинности всех атомов формулы называют ее интерпретацией. Пусть формула P разрешенная формула без свободных переменных. Интерпретацияформулы P определяется следующим образом.

f2. Если $P=\neg G$, то в G нет свободных переменных. Формула P истинная, когда G ложная, и ложная, когда формула G истинная.

f3. Если $P=G\wedge Q$ или $P=G\vee Q$, то в G или Q нет свободных переменных. Если $P=G\wedge Q$, тоформула P истинная, тогда когда формулы G и Q обе истинные, в противном случае формула P ложная. Если $P=G\vee Q$, тоформула P ложная, тогда когдаформулы G и Q обе ложные, в противном случае формула P истинная.

f4. Если $P=\exists\boldsymbol{x}(R)G$, то $\boldsymbol{x}$ – единственная свободно входящая в формулу G переменная. Формула P истинная, если существует строка схемы R, $s\in S(R)$, такая, что формула $G(s/\boldsymbol{x})$ – истинная, иначе формула P ложная.

f5. Если $P=\forall\boldsymbol{x}(R)G$, то $\boldsymbol{x}$ – единственная свободно входящая в формулу G переменная. Формулаистинная, если для каждой строки схемы R, $s\in S(R)$ формула $G(s/\boldsymbol{x})$ истинная, иначе формула P ложная.

f6. Если $P=(G)$, то формула P истинная, если формула G истинная, и ложная, если формула G ложная.

Пусть $E=\{\boldsymbol{x}(R)\,|\,P(\boldsymbol{x})\}$ – выражение исчисления строк. Значением выражения E назовем таблицу схемы R, содержащую все строки $s\in S(R)$, такие, что формула $P(s/\boldsymbol{x})$ истинная.

*Теорема***.** Если E – выражение табличной алгебры, то можно эффективно (и равномерно) построить эквивалентное ему выражение F исчисления строк.

Доказательство. При доказательстве теоремы рассматриваем выражения табличной алгебры, которые содержат только операции объединения, пересечения, разницы, селекции, проекции, соединения и переименования потому, что (оставшиеся) операции деления и активного дополнение можно выразить через данные операции: $E_1\div^{R_1}_{R_2}E_2=\pi_{R'}(E_1)\setminus_{R'}\pi_{R'}(\pi_{R'}(E_1)\otimes E_2\setminus_{R_1}E_1)$, где $R_2\subseteq R_1$, $R'=R_1\setminus R_2$;

$\widetilde{E}_1 = C(E_1) \setminus E_1$, где $C(E_1) = \pi_{A_1}(E_1) \otimes ... \otimes \pi_{A_n}(E_1)$, $R = \{A_1,...,A_n\}$ – схема таблицы, которая является значением выражения E_1 (см., например, [1, 4]).

Доказательство будем проводить индукцией по числу операций в E.

Базис индукции (нет операций). Имеют место два случая. $E = t$, где t – таблица схемы R, положим $F = \{\boldsymbol{x}(R) \mid t(\boldsymbol{x})\}$. E – постоянная таблица $t = \{s_1,...,s_n\}$ схемы $\{A_1,...,A_m\}$, причем $s_i = \left\{\left\langle A_1, d_{i_1}\right\rangle,...,\left\langle A_m, d_{i_m}\right\rangle\right\} = \bigcup_{j=1}^{m}\left\{\left\langle A_m, d_{i_j}\right\rangle\right\}$. Положим $F = \{\boldsymbol{x}(\{A_1,...,A_m\}) \mid \bigvee_{i=1}^{n} \bigwedge_{j=1}^{m} x(A_j) = d_{i_j}\}$. □

Шаг индукции. Предположим, что утверждение теоремы справедливо для любого выражения табличной алгебры, содержащего менее k операций. Пусть выражение E содержит k операций.

Случай 1(объединение). $E = E_1 \cup_R E_2$. Тогда E_1 и E_2 имеют менее k операций, поэтому существуют выражения исчисления строк $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x})\}$ и $\{\boldsymbol{x}(R) \mid Q(\boldsymbol{x})\}$, эквивалентные E_1 и E_2 соответственно. Положим F равным $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x}) \vee Q(\boldsymbol{x})\}$. □

Случай 2(разность). $E = E_1 \setminus_R E_2$. Тогда как и в случае 1 существуют выражения исчисления строк $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x})\}$ и $\{\boldsymbol{x}(R) \mid Q(\boldsymbol{x})\}$, эквивалентные E_1 и E_2 соответственно. Положим F равным $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x}) \wedge \neg Q(\boldsymbol{x})\}$. □

Случай 3(пересечение). $E = E_1 \cap_R E_2$. Тогда существуют выражения исчисления строк $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x})\}$ и $\{\boldsymbol{x}(R) \mid Q(\boldsymbol{x})\}$, эквивалентные E_1 и E_2 соответственно. Положим F равным $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x}) \wedge Q(\boldsymbol{x})\}$. □

Случай 4(селекция). $E = \sigma_{\widetilde{p},R}(E_1)$. Пусть $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x})\}$ – выражение исчисления строк, эквивалентное E_1. Положим F равным $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x}) \wedge p(\boldsymbol{x}(A_1),..., \boldsymbol{x}(A_m))\}$, где $R = \{A_1,..., A_m\}$ схема таблицы, которая есть значением выражения E_1. Здесь предполагается, что предикат-параметр селекции задан как $\widetilde{p}(s) = T \Leftrightarrow \boldsymbol{p}(s(A_1),...,s(A_m)) = T$, $s \in S(R)$, где $\boldsymbol{p}$ – m-арный сигнатурный предикатный символ. □

Случай 5(проекция). $E = \pi_{X,R}(E_1)$. Пусть $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x})\}$ – выражение исчисления строк, эквивалентное E_1. Положим F равным $\{\boldsymbol{y}(X \cap R) \mid \exists \boldsymbol{x}(R)(P(\boldsymbol{x}) \wedge \bigwedge_{A \in X \cap R} \boldsymbol{y}(A) = \boldsymbol{x}(A))\}$. □

Случай 6(соединение). $E = E_1 \underset{R_1,R_2}{\otimes} E_2$. Пусть $\{\boldsymbol{x}(R_1) \mid P(\boldsymbol{x})\}$ и $\{\boldsymbol{y}(R_2) \mid Q(\boldsymbol{y})\}$ – выражения исчисления строк, эквивалентные E_1 и E_2 соответственно. Положим F равным $\{\boldsymbol{z}(R_1 \cup R_2) \mid \exists \boldsymbol{x}(R_1) \exists \boldsymbol{y}(R_2)(P(\boldsymbol{x}) \wedge Q(\boldsymbol{y}) \wedge \bigwedge_{A \in R_1} \boldsymbol{z}(A) = \boldsymbol{x}(A) \wedge \bigwedge_{A \in R_2} \boldsymbol{z}(A) = \boldsymbol{y}(A))\}$. □

Случай 7(переименование). $E = Rt_{\xi,R}(E_1)$, где $\xi : \boldsymbol{A} \xrightarrow{\sim} \boldsymbol{A}$ – инъективная функция переименования атрибутов. Тогда существует выражение исчисления строк $\{\boldsymbol{x}(R) \mid P(\boldsymbol{x})\}$, эквивалентное E_1. Положим F равным $\{\boldsymbol{y}(R_2) \mid \exists \boldsymbol{x}(R)(P(\boldsymbol{x}) \wedge \bigwedge_{C \in R \setminus dom\xi} \boldsymbol{y}(C) = \boldsymbol{x}(C) \wedge \bigwedge_{A \in R \cap dom\xi} \boldsymbol{x}(A) = \boldsymbol{y}(\xi(A)))\}$, где $R_2 = R \setminus dom\xi \cup \xi[R]$. □ ◘

Выводы

В роботе проведено обобщение табличной алгебры на таблицы, которым приписывается определенная схема. Заданы основные операции над обобщенными таблицами. Классическое исчисления строк пополнено произвольными предикатными и функциональными сигнатурами на универсальном домене (в то время как обычно рассматривают лишь бинарные предикаты, а функциональная сигнатура вообще пуста). Определено синтаксис термов, атомов и формул исчисления строк; выделен класс разрешенных формул, используя понятие свободных и связанных переменных строк, введено понятие схемы $scheme(x,P)$ и множества атрибутов $attr(x,P)$, с которыми переменная строка встречается в формулах. Доказано, что исчисление строк остается не менее выразительным, чем табличная алгебра. Задача для последующей работы установить дуальный результат.

Литература

[Редько, Брона, Буй, Поляков, 2001] Реляционные базы данных: табличные алгебры и SQL-подобные языки / В. Н. Редько, Ю. Й. Брона, Д. Б. Буй, С. А. Поляков. Киев: Издательский дом «Академпериодика», 2001.

[Codd, 1972] E. F. Codd. Relational Completeness of Data Base Sublanguages. In: Data Base Systems. New York: Prentice-Hall, 1972. – P. 65-93.

[Lacroix, Pirotte, 1977] M. Lacroix, A. Pirotte. Domain-oriented relational languages / M. Lacroix, A. Pirotte. In: Proc. 3rd Int. Conf on Very Large Data Bases. Tokyo, October, 1977. – P. 370-378.

[Мейєр, 1977] Д. Мейєр. Теория реляционных баз данных. Москва: Мир,1987.

[Ульман, 1977] Дж. Ульман. Основы систем баз данных. Москва: Финансы и статистика, 1983.

[Дейт, 2005] К. Дж. Дейт. Введение в системы баз данных: [8-е изд.: пер. с англ.]. Москва: Издательский дом «Вильямс», 2005.

[Коннолли, Бегг, 2003] Т. Коннолли, К. Бегг. Базы данных. Проектирование, реализация и сопровождение. Теория и практика: [3-изд.: пер. с англ.]. Москва: Издательский дом «Вильямс», 2003.

[Пасичник, Резниченко, 1977] В. В. Пасичник, В. А Резниченко. Организация баз данных и знаний. Киев: Издательская группа BHV, 2006.

[Мендельсон, 1977] Э.Мендельсон. Введение в математическую логику. Москва: Наука, 1971.

[Цаленко, 1977] М. Ш. Цаленко. Моделирование семантики в базах данных. Москва: Наука,1989.

Информация об авторах

***Дмитрий Буй** – доктор физико-математических наук, старший научный сотрудник, Киевский национальный университет имени Тараса Шевченко, факультет кибернетики: Украина, Киев, 03680, пр. Глушкова 2, корп. 6; e-mail: buy@unicyb.kiev.ua.*

***Ирина Глушко** – аспирантка, Киевский национальный университет имени Тараса Шевченко, факультет кибернетики: Украина, Киев, 03680, пр. Глушкова 2, корп. 6; e-mail: glushkoim@gmail.com.*