ISSN 0136-3549
0320-3409

TALLINNA
POLÜTEHNILISE INSTITUUDI
TOIMETISED

464

ТРУДЫ ТАЛЛИНСКОГО
ПОЛИТЕХНИЧЕСКОГО
ИНСТИТУТА

TPI '79

АНАЛИЗ ДАННЫХ. ПОСТРОЕНИЕ ТРАНСЛЯТОРОВ. ВОПРОСЫ ПРОГРАММИРОВАНИЯ

Труды экономического
факультета XXXУII

464

TALLINNA POLÜTEHNILISE INSTITUUDI TOIMETISED

ТРУДЫ ТАЛЛИНСКОГО ПОЛИТЕХНИЧЕСКОГО ИНСТИТУТА

TPI '79

УДК 51:801; 519.24; 681.3.016:001.5; 512.25/26+519.3:330.115, 518.9:65.012, 681.3.016

АНАЛИЗ ДАННЫХ. ПОСТРОЕНИЕ ТРАНСЛЯТОРОВ. ВОПРОСЫ ПРОГРАММИРОВАНИЯ

Труды экономического факультета XXXУ11

Таллин 1979

№ 464

TALLINNA POLÜTEHNILISE INSTITUUDI TOIMETISED

ТРУДЫ ТАЛЛИНСКОГО ПОЛИТЕХНИЧЕСКОГО ИНСТИТУТА

УДК 51:801

М.В.Лепп, А.О.Вооглайд

ОПЫТ ВНЕДРЕНИЯ КЛАССИЧЕСКИХ МЕТОДОВ СИНТАКСИЧЕСКОГО АНАЛИЗА

Введение

Настоящая статья предназначена пользователям Системы построения трансляторов (СПТ). Выражаясь словами И.Лекарма [7], пользователь СПТ - это опытный специалист по системному программированию или старший программист, или выпускник факультета системного программирования, желающий создать транслятор для старого или нового специализированного языка программирования средних размеров. Под СПТ подразумевается комплекс теоретических методов, технических приемов и программное обеспечение, при помощи которого можно технологизировать и автоматизировать процесс изготовления транслятора и его отдельных частей [6].

При производстве трансляторов наиболее полно автоматизированным процессом является создание синтаксического анализатора, который и определяет эффективность и природу составляемого транслятора.

В статье рассматриваются проблемы синтаксического анализа и некоторые аспекты семантического анализа. Область синтаксиса языков программирования теоретически разработана достаточно глубоко. Поэтому проблема синтаксиса часто считается как бы решенной.

Однако при внедрении теории в практику возникают новые проблемы, важнейшей из которых является проблема эффективности и достижения метода синтаксического анализа, удобного с точки зрения инженерной психологии. Этот вопрос возникает особо остро при работе на ЭВМ средней мощности ("Минск-32", ЕС-1020 - ЕС-1030).

Опираясь на многолетний опыт работы в области автоматизации производства трансляторов на разных ЭВМ, авторы статьи предлагают новый метод синтаксического анализа типа восходящего разбора, реализация которого обеспечивает достаточную эффективность синтаксического анализатора и облегчает проведение семантического анализа.

В действительности же как обсуждаемые здесь, так и многие другие проблемы возникают на более раннем этапе, а именно, уже при выборе метода анализа.

От выбора метода зависит, насколько удобно пользователю описывать создаваемый язык, каким образом возможно связывать между собой семантические действия, какого объема работ требует внесение изменений и исправлений в уже созданный транслятор, какова будет скорость получаемого анализатора и необходимый для него объем памяти. Выбор метода определяет также такие свойства анализатора, как степень стыковки СПТ с операционной системой машины и существование средств диагностики и исправления синтаксических ошибок. Причем, с точки зрения практики можно говорить не только об одном трансляторе, но о системе программирования. Из всего вышесказанного вытекает, что выбранный метод должен быть одновременно и универсальным и эффективным. Достичь этого довольно трудно.

Обычно обобщенные методы, с одной стороны, обеспечивают пользователя всем необходимым для удобного использования системы, а с другой стороны, требуют больших ресурсов памяти и значительной скорости машины.

Если метод эффективен, то он обязательно базируется на каком-либо специальном классе грамматик, налагающем существенные ограничения на правила синтаксиса. Возникающее противоречие невозможно преодолеть, если в основу метода положен только один класс грамматик. Противоречие преодолевается, если иметь минимально две грамматики, генерирующие один и тот же язык. Первая из них, которая относительно свободна от ограничений, предназначена для описания создаваемого языка и осуществляет стыковку с семантикой. Вторая грамматика, которая налагает существенные ограничения на множество правил в роли распознавателя слов языка (программ, написанных на исходном языке).

В таком случае пользователь может оперировать только общей грамматикой, не считаясь с грамматикой, используемой для распознавания слов языка. Система же должна гарантировать идентичность синтаксических деревьев [5,12] каждого слова, образованного по первой и второй грамматикам.

При нашей системе выбора грамматик пользователю удобно описывать создаваемый язык, и полученный в результате синтаксический анализатор становится более эффективным. Для описания выбирается грамматика, похожая на грамматику в формулах Бэкуса-Наура [3]. Например, в ТПИ для этого используют формулы Бэкуса-Наура с расширениями Лукаса [13].

При анализе же используются грамматики, которые наиболее эффективны и пригодны для реализации на машинах средней мощности.

Чтобы получить эффективный метод анализа, следует стремиться еще до проведения анализа определить самый подходящий метод для каждого шага анализа. Для этого ни один классический метод в чистом виде не подходит.

Поэтому в первой части статьи предлагается новый эффективный метод восходящего разбора, допускающий проведение ряда работ синтаксического анализа еще во время генерирования анализатора.

Во второй части статьи демонстрируется возможность использования данного метода как средства проведения некоторых работ семантического анализа.

I. Метод синтаксического анализа, работающий в каскадной технике

I. Основные понятия

Обозначим через $\mathcal{A}^*$ множество всех слов в алфавите $\mathcal{A}$ и $\mathcal{A}^+ = \mathcal{A}^* \setminus \{\lambda\}$, где λ - пустое слово. Если $x \in \mathcal{A}^*$, то $|x|$ - длина слова x. Контекстно-свободная грамматика (КГС) - это упорядоченная четверка $G = (\mathcal{V}_N, \mathcal{V}_T, P, S)$, где $\mathcal{V}_N$ - конечный алфавит нетерминальных символов, $\mathcal{V}_T$ - конечный алфавит терминальных символов такой, что $V_N \cap V_T = \emptyset$, P – множество правил подстановки вида $A \to x$, где $A \in V_N$, $x \in (V_N \cup V_T)^*$ и $S (S \in V_N)$ – начальный символ или аксиома. Обозначим $V = V_N \cup V_T$. Пусть $x, y \in V^*$. Слово y непосредственно канонически выводимо из слова $x (x \Rightarrow y)$, если $x = uAv$, $y = uzv$ и $A \to z \in P$, $u \in V^*$, $v \in V_T^*$. Слово y канонически выводимо из слова $x (x \overset{*}{\Rightarrow} y)$, если существует последовательность слов $z_0, z_1, \ldots, z_k (z_i \in V^*, 0 \leq i \leq k)$ такая, что $x = z_0$, $y = z_k$ и $z_{i-1} \Rightarrow z_i (0 < i \leq k)$. Последовательность $z_0, z_1, \ldots, z_k$ называется каноническим выводом y из x. Слово $x \in V^*$ называется канонической сентенциальной формой, если $S \overset{*}{\Rightarrow} x$. Языком $\mathcal{L}$, определяемым грамматикой G, называется множество $\mathcal{L}(G) = \{x \mid x \in V_T^* \,\&\, S \overset{*}{\Rightarrow} x\}$.

Если $S \overset{*}{\Rightarrow} uAv \Rightarrow uzv$ и $A \to z \in P$, то слово z называется основой канонической сентенциальной формы uzv.

Каноническим разбором слова $x \in V^*$ называется последовательность слов $z_k, z_{k-1}, \ldots, z_1, z_0$, такая, что $z_k = x$, $z_0 = S$ и последовательность $z_0, z_1, \ldots, z_k$ является каноническим выводом слова x.

В дальнейшем мы опустим понятие "канонический".

КСГ $G = (V_N, V_T, P, S)$ называется λ-свободной, если все правила из P имеют вид $A \to x$, $x \in V^+$. КСГ G называется приведенной, если для каждого нетерминала $A (A \neq S)$ существуют слова $x, y \in V^*$ и $z \in V_T^*$ такие, что $S \overset{*}{\Rightarrow} xAy$ и $A \overset{*}{\Rightarrow} z$ и в грамматике G нет выводов $A \overset{+}{\Rightarrow} A$ для $A \in V_N$. КСГ называется обратимой, если множество P не содержит правил вида $A \to x$ и $B \to x$, где $A \neq B$. Конфликтом редуцирования называется тройка (A, B, x), где $A \to x, B \to x \in P$ и $A \neq B$.

Пусть $\mathcal{K}$ -некоторый класс КС-грамматик. Мы будем говорить, что язык L принадлежит к классу $\mathcal{K}$, если $L = \mathcal{L}(G)$ для некоторого $G \in \mathcal{K}$.

2. Используемые на практике восходящие методы анализа

За основу конструирования метода синтаксического анализа, который приводится в следующем пункте, принимаются статья [II] и три классических метода синтаксического анализа:

метод предшествования;

метод ограниченного канонического контекста;

метод смешанной стратегии предшествования.

Хотя понятия и результаты, связанные с поименованными методами представлены в монографии [5], все же для обеспечения восприятия они приводятся в этом пункте статьи.

Определение I. Пусть $G(V_N, V_T, P, S)$ - КС-грамматика. Отношениями предшествования $\lessdot, \doteq, \gtrdot$ называются следующие отношения:

(I) $X \lessdot Y$, если найдется правило $A \to uXBv \in P$ такое, что $B \overset{+}{\Rightarrow} Yz \, (X, Y \in V, u, v, z \in V^*)$;

(2) $X \doteq Y$, если найдется правило $A \to uXYv \in P \, (X, Y \in V, u, v \in V^*)$;

(3) $X \gtrdot T$, если найдется правило $A \to uBCv$ такое, что $B \overset{+}{\Rightarrow} z_1 X$ и $C \overset{*}{\Rightarrow} T z_2 \, (X \in V, T \in V_T, u, v, z_1, z_2 \in V^*)$.

Определение 2. λ - свободная КС-грамматика называется грамматикой предшествования, если $\lessdot, \doteq, \gtrdot$ попарно не пересекаются.

При помощи отношений предшествования можно локализировать основу в сентенциальной форме. Это видно из следующей теоремы [2].

Теорема I. Пусть $G = (V_N, V_T, P, S)$ грамматика предшествования и $S \overset{*}{\Rightarrow} X_1 X_2 \ldots X_K A T_1 T_2 \ldots T_\ell \Rightarrow X_1 X_2 \ldots X_K Y_1 Y_2 \ldots Y_m T_1 T_2 \ldots T_\ell$ в G, то

(I) $X_i \lessdot X_{i+1} \vee X_i \doteq X_{i+1} \; (1 \leq i < K)$;

(2) $X_K \lessdot Y_1$;

(3) $Y_j \doteq Y_{j+1} \; (1 \leq j < m)$;

(4) $Y_m \gtrdot T_1$;

(5) никакие другие отношения предшествования, кроме указанных в пунктах (I)-(4), между этими символами не имеют места.

Определение 3. Обратная грамматика предшествования называется грамматикой простого предшествования.

Обозначим класс грамматик простого предшествования через $\mathcal{K}_1$.

Слово языка, порожденное грамматикой простого предшествования, очень легко распознаваемо, так как основа точно локализируема с помощью отношений предшествования, и для каждой основы существует в точности одно правило из P, при помощи которого можно осуществить редуцирование. Поэтому и анализаторы, работающие на классе грамматик простого предшествования, являются максимально эффективными. К сожалению, грамматики простого предшествования для практических целей часто неприемлемы из-за малой мощности порождаемости. Существует много детерминированных КС-языков, которые не генерируются грамматиками простого предшествования.

Определение 4. КС-грамматика G называется грамматикой слабого предшествования, если:

I) отношение $\gtrdot$ не пересекается с отношениями $\lessdot$ и $\doteq$;

2) если существуют правила $A \rightarrow uXv$ и $B \rightarrow v$, то не имеет место $X \in \{Y \mid S \overset{*}{\Rightarrow} xYBy,\ x,y \in V^*,\ Y \in V\}$.

Из этого определения видно, что при локализации основы в сентенциальной форме uzv с помощью отношений предшествования можно точно локализировать только правый конец основы (условие I). Второе условие гарантирует, что редуцирование осуществимо только при помощи самой длинной продукции в тех случаях, где $A \rightarrow uz$ и $A \rightarrow z$ принадлежат множеству продукции.

Порождающая мощность как грамматик слабого предшествования, так и грамматик простого предшествования совпадает.

Теорема 2. Язык определяется обратимой грамматикой слабого предшествования тогда и только тогда, когда он является языком простого предшествования [4].

Можно расширить класс генерируемых языков, используя в роли порождающих грамматик грамматики (m, k) -ограниченного канонического контекста.

Определение 5. КС-грамматика $G=(V_N, V_T, P, S)$ называется грамматикой с (m, k) -ограниченным каноническим контекстом, если из условий $\#^m S \#^k \stackrel{*}{\Rightarrow} u_1 x_1 A_1 y_1 v_1 \Rightarrow u_1 x_1 z_1 y_1 v_1$ и $\#^m S \#^k \stackrel{*}{\Rightarrow} u_2 x_2 A_2 y_2 v_2 \Rightarrow u_2 x_2 z_2 y_2 v_2 = u_3 x_1 z_1 y_1 v_3$,

где $u_i, x_j, z_j \in V^*$; $A_j \in V_N$; $v_i, y_j \in V_T^*$; $|x_j| = m$; $|y_j| = k$
$(i = 1,2,3;\ j = 1,2)$,

всегда следует, что

$$u_2 = u_3,\ x_1 = x_2,\ z_1 = z_2,\ y_1 = y_2,\ v_2 = v_3 \text{ и } A_1 = A_2.$$

Сущность метода анализа, базирующегося на классе грамматик (m, k) -ограниченного канонического контекста, состоит в том, что шаг анализа осуществим только в ходе просмотра m -символьного контекста слева от основы и k -символьный контекст - справа от нее. Это ведет к тому, что во время анализа для каждого нетерминала необходимы множество (m, k) - контекста и функции решения для определения действий, зависимых от контекста. При сравнении по эффективности анализаторов простого предшествования и (m, k) - ограниченного канонического контекста становится очевидным, что анализаторы простого предшествования существенно эффективнее.

Класс грамматик (m, k) - ограниченного канонического контекста имеет свои преимущества. Порождающая мощность грамматик этого класса $(m > 0)$ позволяет генерировать все детерминированные языки [4].

На базе грамматик слабого предшествования в статье [I] предложен метод смешанной стратегии предшествования.

Определение 6. (m, k) -ограниченным каноническим контекстом (сокращенно (m, k) -ОКК) нетерминала A называется множество пар (обозначим $C_A^{m,k}$):

$$C_A^{m,k} = \{(x, y) \mid \#^m S \#^k \stackrel{*}{\Rightarrow} u x A y v;\ u, x \in V^*,\ |x| = m, |y| = k\}.$$

Введем следующее обозначение $C_A^{m/k} = C_A^{m,0} \times C_A^{0,k}$.

Определение 7. Пусть $G = (V_N, V_T, P, S)$ - приведенная КСГ. G называется грамматикой смешанной стратегии предшествования, если выполняются следующие условия:

I) отношение $\cdot>$ не пересекается с отношениями $<\cdot$ и $\doteq$;

2) если $A \to uX\nu$ и $B \to \nu$ принадлежат множеству P, то $X \notin C_B^{1,0}$;

3) если множество P содержит конфликт редуцирования (A, B, x), то $C_A^{1,0} \cap C_B^{1,0} = \emptyset$.

Обозначим класс обратимых грамматик слабого предшествования через $\mathcal{K}'_1$, и класс грамматик смешанной стратегии предшествования – через $\mathcal{K}'_2$.

Обобщим грамматику смешанной стратегии предшествования, заменив в условии 3) требование $C_A^{1,0} \cap C_B^{1,0} = \emptyset$ требованием $C_A^{1/1} \cap C_B^{1/1} = \emptyset$ или $C_A^{1,1} \cap C_B^{1,1} = \emptyset$. Так получаем еще два класса грамматик $\mathcal{K}'_3$ и $\mathcal{K}'_4$.

Каждый из классов грамматик $\mathcal{K}'_2, \mathcal{K}'_3, \mathcal{K}'_4$ порождает в точности класс детерминированных контекстно-свободных языков [I], поскольку $\mathcal{K}'_1 \subset \mathcal{K}'_2 \subset \mathcal{K}'_3 \subset \mathcal{K}'_4$.

Далее приводится аналогичным образом определение нового класса грамматик, который вобрал бы в себя эффективность анализа грамматик простого предшествования и способность определять все детерминированные языки.

Следуя статье [2], разобьем шаг анализа на два действия:

I. Детектирование. Найти основу z в сентенциальной форме $uz\nu$.

2. Редуцирование. Найти правило $A \to z$ такое, что $S \overset{*}{\Rightarrow} uA\nu \Rightarrow uz\nu$ и заменить основу z на A.

Используя для детектирования отношения предшествования, а для редуцирования – (m,k)-ОКК, определяем новый класс грамматик [II].

Определение 8. Если G – КС-грамматика, то G называется (m,k)-ОКК редуцируемой грамматикой предшествования (ОККРГП), если G -грамматика предшествования и $C_A^{m,k} \cap C_B^{m,k} = \emptyset$ при (A, B, x).

С точки зрения реализации представляют интерес ОККРГП если $m,k \leqslant 1$. В этом случае эффективность соответствующего анализатора сравнима с эффективностью анализатора простого предшествования. А порождающая мощность классов грамматик (I,0-ОККРГП, (I/I)-ОККРГП и (I,I)-ОККРГП позволяет определить все детерминированные языки [II].

Обозначим (0,0)-ОККРГП через $\mathcal{K}_1$,

(I,0)-ОККРГП через $\mathcal{K}_2$,

(I/I)-ОККРГП через $\mathcal{K}_3$,

(I,I)-ОККРГП через $\mathcal{K}_4$.

3. Метод каскада

В основу конструирования анализатора, работающего в каскадной технике, положены следующие классы грамматик:

$$\mathcal{K}_1, \dots, \mathcal{K}_4, \mathcal{K}'_1, \dots, \mathcal{K}'_4.$$

Уже отмечалось, что эффективность анализаторов, работающих на базе классов грамматик $\mathcal{K}_2$, $\mathcal{K}_3$ или $\mathcal{K}_4$, сравнима с эффективностью анализатора класса $\mathcal{K}_1$. На самом деле они почти совпадают, так как части анализаторов, зависящие от языка (матрицы предшествования), совпадают, а инвариантные части анализаторов отличаются только в отношении работы в местах конфликтов редуцирования. Дополнительная работа в анализаторах классов $\mathcal{K}_3$ и $\mathcal{K}_4$ сравнима с умножением двух булевских векторов. Однако анализатор класса грамматик $\mathcal{K}_4$ требует дополнительных таблиц канонического контекста. Осложняется и алгоритм разбора при редуцировании. Несмотря на то, что анализаторы классов грамматик $\mathcal{K}_4$ требуют большего объема памяти и более длительного времени работы машины, чем анализаторы классов грамматик $\mathcal{K}_1$, $\mathcal{K}_2$ или $\mathcal{K}_3$, не целесообразно отказываться от класса грамматик $\mathcal{K}_4$. Причина, не позволяющая отказаться от него, кроется в семантическом анализе. Дело в том, что, если необходимо преобразовать произвольную грамматику в грамматику, принадлежащую классам $\mathcal{K}_1$, $\mathcal{K}_2$ или $\mathcal{K}_3$, то в местах, которые в исходной грамматике анализировались с помощью LR(к)-контекста, происходит кардинальное изменение структуры, что значительно осложняет семантически анализ. Как показывает практика, при игнорировании класса грамматик $\mathcal{K}_4$ потери гораздо больше, нежели выигрыши.

В статье [II] доказаны следующие отношения включения: $\mathcal{K}_1 \subset \mathcal{K}_2 \subset \mathcal{K}_3 \subset \mathcal{K}_4$. Взяв за основу для каждого шага анализа последнее обстоятельство, можно определить технику осуществления этого шага. Причем, данная техника определяется следующим образом. Если шаг анализа осуществляется в технике, соответствующей классу грамматик $\mathcal{K}_i$ $(1 < i \leq 4)$, то этот шаг невозможно реализовать в технике, соответствующей классу грамматик $\mathcal{K}_{i-1}$. В этом и заключается метод каскада, используемый для синтаксического анализа.

Во время генерирования синтаксического анализатора для каждого шага анализа можно выбрать технику с меньшим количеством работ. Так как при классах грамматик $\mathcal{K}_1, \ldots, \mathcal{K}_4$ для детектирования используется метод предшествования, то применение метода каскада состоит в выборе техники редуцирования.

С каждой основой связывается одна грамматика из множества $\{\mathcal{K}_1, \ldots, \mathcal{K}_4\}$ таким образом, чтобы основа была редуцируема при помощи грамматики $\mathcal{K}_i$ $(1 < i \leq 4)$ и не редуцируема при помощи грамматики $\mathcal{K}_{i-1}$.

Нужное разбиение множества продукций на непересекающиеся классы эквивалентности получаем следующим образом.

Пусть w_x -множество пар (A, B), где $A \neq B$ и $A \to x, B \to x$ принадлежат множеству продукций P,

$$w_x = \{(A, B) \mid A \neq B \,\&\, A \to x, B \to x \in P\}.$$

Разделим множество продукции на четыре класса следующим образом:

$$P_1 = \{X \to x \in P \mid w_x = \phi\};$$

$$P_2 = \{X \to x \in P \mid ((A,B) \in w_x) \,\&\, C_A^{1,0} \cap C_B^{1,0} = \phi\};$$

$$P_3 = \{X \to x \in P \mid ((A,B) \in w_x) \,\&\, (C_A^{1,0} \cap C_B^{1,0} \neq \phi) \,\& \\ C_A^{1/1} \cap C_B^{1/1} = \phi\},$$

$$P_4 = \{X \to x \in P \mid ((A\ B) \in w) \,\&\, (C_A^{1/1} \cap C_B^{1/1} \neq \phi) \,\& \\ (C_A^{1,1} \cap C_B^{1,1} = \phi)\}.$$

Отсюда видно, что

$$P_i \cap P_j = \phi,\ i \neq j \quad \text{и} \quad P_1 \cup P_2 \cup P_3 \cup P_4 = P.$$

Пример. Пусть дана грамматика $G=(V_N, V_T, P, S)$, где множество P состоит из следующих правил:

$$
\begin{array}{llll}
P: & S \to aA & & C \to 1A_2 0 \\
 & S \to bB & & C \to 0A_2 0 \\
 & A \to C & & A_1 \to d \\
 & B \to C & & A_2 \to d \\
 & C \to 0A_1 1 & &
\end{array}
$$

Тогда

$$
\begin{array}{llll}
P_1: & S \to aA & & C \to 1A_2 0 \\
 & S \to bB & & C \to 0A_2 0, \\
 & C \to 0A_1 1 & & \\
P_2: & A \to C & & \\
 & B \to C, & &
\end{array}
$$

так как

$$C_A^{1,0}=\{a\},\quad C_B^{1,0}=\{b\} \quad \text{и} \quad C_A^{1,0}\cap C_B^{1,0}=\phi .$$

$$
\begin{array}{ll}
P_3: & A_1 \to d \\
 & A_2 \to d
\end{array}
$$

так как из $C_{A_1}^{1,0}=\{0\}$ и $C_{A_2}^{1,0}=\{0,1\}$ следует, что $C_{A_1}^{1,0}\cap C_{A_2}^{1,0}\neq\phi$, но из $C_{A_1}^{1/1}=\{(0,1)\}$ и $C_{A_2}^{1/1}=\{(1,0)(0,0)\}$ следует, что $C_{A_1}^{1/1}\cap C_{A_2}^{1/1}=\phi$.

Рассмотрим разбор слова $b0d0$. Во-первых, нужно редуцировать основу d. Зная, что продукции с правой частью d принадлежат классу P_3, редуцирование надо осуществлять при помощи (I/I)-канонического контекста. Соответственно, элемент (0,0) удовлетворяет только продукции $A_2 \to d$. В результате этого редуцирования получаем сентенциальную форму $b0A_20$, где основой является $0A_20$.

Продукция с правой частью $0A_20$ принадлежит классу P_1. Значит, основу можно непосредственно редуцировать. Процесс продолжается аналогично до конца.

Обозначим синтаксический анализатор, работающий методом каскада на базе класса грамматик $\mathcal{K}_i$ $(1 \leqslant i \leqslant 4)$, через A.

Исходим из классов грамматик $\mathcal{K}'_1,\ldots,\mathcal{K}'_4$ и отношений $\mathcal{K}'_1 \subset \mathcal{K}'_2 \subset \mathcal{K}'_3 \subset \mathcal{K}'_4$. Легко убедиться в правильности последних отношений.

Отношения $\mathcal{K}'_i \subseteq \mathcal{K}'_{i+1}$ $(1 \leqslant i < 4)$ вытекают непосредственно из определений самих этих классов. Для доказательства, что $\mathcal{K}'_i \subset \mathcal{K}'_{i+1} (1 \leqslant i < 4)$, достаточно нахождения такой грамматики G, что $G \in \mathcal{K}'_{i+1}$ и $G \notin \mathcal{K}'_i$.

Учитывая принцип работы анализатора A, аналогичным образом на базе классов $\mathcal{K}'_i$ $(1 \leqslant i \leqslant 4)$ можно образовать синтаксический анализатор, в котором для каждого шага анализа определена техника с наименьшим количеством работ.

Причем, особо следует подчеркнуть преимущество метода каскадной техники перед классическими методами. Он дает возможность проведения ряда необходимых для анализа работ в ходе генерирования синтаксического анализатора, тем самым значительно сократить затрачиваемое на конкретный разбор время.

II Синтаксический анализ как одно из средств проведения семантического анализа

Как правило, все этапы работы транслятора строго разграничиваются (например, лексический анализ, синтаксический анализ, преобразование синтаксического дерева, инициализация семантических атрибутов, генерирование текста объектного языка и т.д.). Такое разграничение необходимо в случае, если объектом исследования служит СПТ как таковая. Совершенно другая ситуация складывается, если СПТ используется в ежедневной работе как средство создания инструментальных систем. В этом случае, особенно при работе на машинах средней мощности, на первый план выдвигается требование эффективности создаваемого транслятора. Именно поэтому не всегда представляется целесообразным придерживаться столь строгого разграничения.

Ниже как раз и рассматриваются некоторые возможности проведения части семантического анализа или во время синтаксического анализа, или его средствами.

Такой подход значительно сокращает время трансляции, что доказывается в заключительной части статьи.

Опираясь на могографию А.Ахо и Дж.Ульмана, можно сказать, что транслятор задан как множество пар (x, y), где x - программа на исходном языке, y - программа на том языке, на который нужно перевести X. Такой перевод рассматривается как преобразование, состоящее из двух частей, вторая из которых условно называется семантической обработкой [5]. Промежуточным языком между этими частями служат, как правило, древовидные структуры [5].

Следовательно, семантической обработкой можно считать преобразование дерева синтаксиса [5] в объектный язык.

Причем, с точки зрения инженерной психологии пользователю удобно свободно определять структуру синтаксического дерева для каждой конкретной конструкции создаваемого языка. С этой целью в ТПИ реализовано в СПТ понятие разреженного дерева ($T_{\mathfrak{D}}^{H,Q}$) [I2].

Приведем некоторые возможности использования синтаксических средств в ходе семантического анализа.

А. Подбор определенной информации. В данной системе предусмотрена возможность определения дополнительных структур на разреженном дереве (с помощью Метаязыка [I3]) в виде счетчиков и списков из элементов $T_{\mathfrak{D}}^{H,Q}$-дерева. Причем, последовательность элементов списка обратная, т.е. является зеркальным отображением последовательности тех самых элементов в $T_{\mathfrak{D}}^{H,Q}$-дереве. Это позволяет произвольно выбрать алгоритм прохождения поддеревьев дерева $T_{\mathfrak{D}}^{H,Q}$.

Б. Отбор сходной информации. В системе предусмотрено деление синтаксических единиц

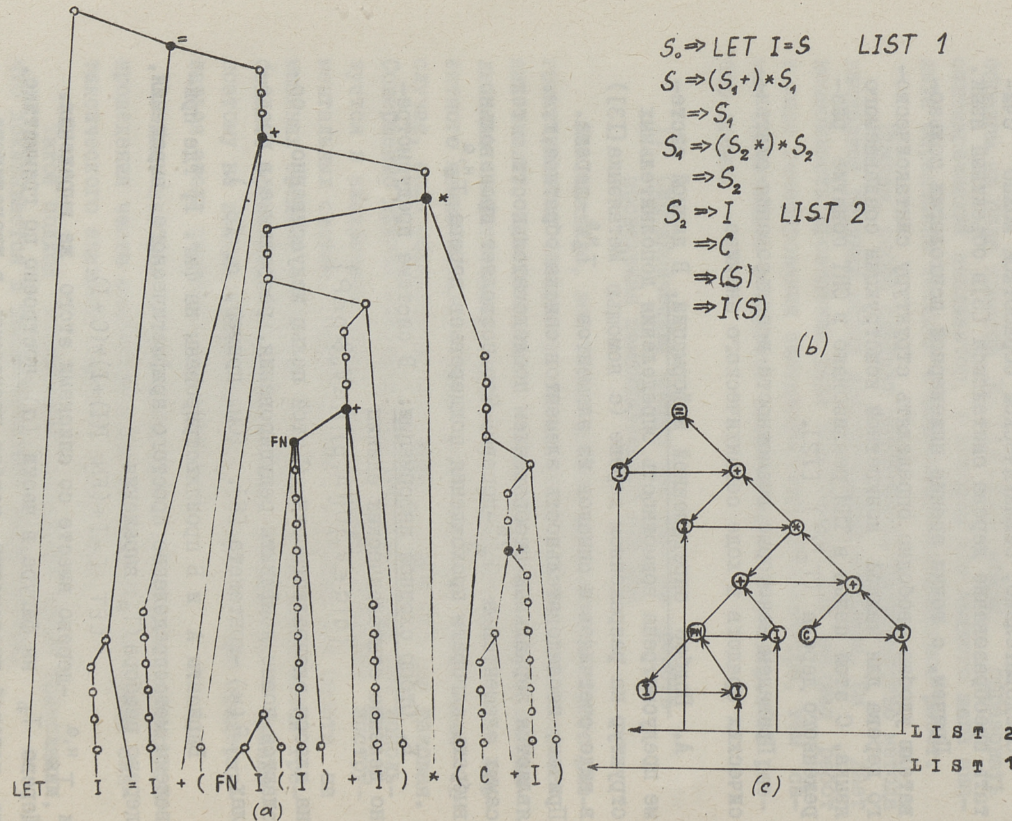
$$\{a \mid a \in V_T\} \cup \{(A,p) \mid A \in V_N \,\&\, p \in P\}$$

на классы эквивалентности. С этой целью искусственно создаются новые конфликты редуцирования, разрешимые в пределах LR(k)-контекста [8]. (См. пример, с. I6).

Варианты А и Б проиллюстрированы на фиг. I, где приведены метаопределение простого арифметического выражения, дерево разбора ($T_{\mathfrak{D}}$) выражения

LET I=I+(FN I(I)+I)*(C+I)

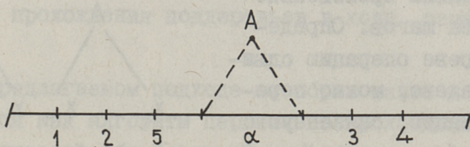
и $T_{\mathfrak{D}}^{H,Q}$ -дерево вместе со списками этого же выражения. Дерево $T_{\mathfrak{D}}$ на фигуре в части (a) построено по грамматике, по которой происходит разбор программы в трансляторе WANG/BASIC ⇒ PL/1 [I0]. Метаопределение (b) является частью метаопределения WANG/BASIC ⇒ PL/1.


LET I = I + (FN I (I) + I) * (C + I)
(a)
$S_0 \Rightarrow LET\ I = S$ LIST 1
$S \Rightarrow (S_1 +) * S_1$
$\Rightarrow S_1$
$S_1 \Rightarrow (S_2 *) * S_2$
$\Rightarrow S_2$
$S_2 \Rightarrow I$ LIST 2
$\Rightarrow C$
$\Rightarrow (S)$
$\Rightarrow I(S)$
(b)
LIST 2
LIST 1
(c)

Фиг. 1.

Списки на разреженном дереве (c) выбраны так, чтобы возможно было найти выражения, в составе которых явно имеются переменные, не приобретающие значений в транслируемой программе.

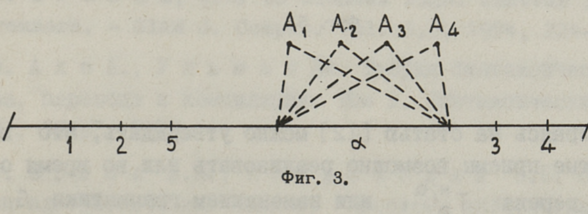
В. <u>Проверка информации с целью конкретизации определенных вариантов семантической обработки.</u> Дадим характеристику этой возможности при помощи примера. Пусть в анализируемой программе имеется конструкция α и пусть ее семантика определена конструкциями в местах I,2,3,4,5. На фигуре это выглядит так:



Фиг. 2.

Пусть корнем поддерева, листьями которого являются элементы конструкции α, будет A. Пусть конструкции α соответствуют четыре различных семантических действия, определенные конструкциями на местах I,2,3,4,5.

Обычно для определения семантического действия в вершине A надо пройти то поддерево дерева $T_{\mathfrak{D}}^{H,Q}$, которое содержит конструкции в местах I,2,3,4,5 и конструкцию α. Если же заменить понятие A соответственно конструкциям в местах I–5 понятиями A_1, A_2, A_3, A_4, то для определения семантики в вершине A нет необходимости в прохождении вышеупомянутого поддерева дерева $T_{\mathfrak{D}}^{H,Q}$.

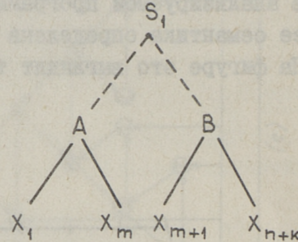


Фиг. 3.

Г. Перенесение информации к месту обработки. Рассмотрим этот вариант на простом примере.

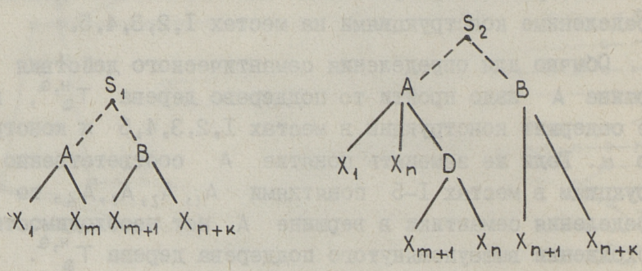
Допустим, что необходимо обработать такую последовательность понятий, как $X_1, \ldots, X_{n+k}$, где часть из них принадлежит к одному поддереву A, а часть – к поддереву B.

Предположим, что для обработки первой части последовательности необходима информация, связанная с понятиями $X_{m+1}, \ldots, X_n$ в поддереве B. Понятия $X_{m+1}, \ldots, X_n$ в поддереве S_1 могут находиться на расстоянии произвольного количества шагов. Определяя в поддереве операцию сдвига основы налево, можно перенести информацию, связанную с вершинами $X_{m+1}, \ldots, X_n$ в поддереве S_1, на один шаг к вершине X_m в поддереве S_2.



Фиг. 4.

Преобразование поддерева S_1 в поддерево S_2 выглядит следующим образом:



Фиг. 5.

Опираясь на статью [I2] можно утверждать, что вышеупомянутые приемы возможно реализовать или во время образования дерева $T^{H,Q}_{\mathfrak{D}}$, или изменением грамматики G при помощи добавления продукции.

Образование дерева $T_{D}^{H,Q}$ проводится аналогично каскадной технике. На каждом шаге анализа точно определено количество работ, необходимое для проведения преобразований дерева. Причем, количество работ, которые выполняются автоматически на каждом шаге анализа, определяется уже во время генерации анализатора.

Добавление продукций в грамматику G не влечет за собой увеличения времени анализа, так как все поиски, связанные со множеством P, проводятся по методу хеширования.

Если не использовать описанные в статье приемы, то перечисленные в пунктах А–Г работы потребовали бы неоднократного прохождения поддеревьев в ходе семантического анализа.

При предлагаемом подходе же образование дерева $T_{D}^{H,Q}$ используется как алгоритм первичного прохождения дерева для семантической обработки. Таким образом значительно сокращается время трансляции.

Литература

1. Aho, A.V., Denning, P.I., Ullman, J.D. Weak and mixed strategy precedence parsing. - J.ACM, 1972, 19, 2,225-243.

2. Gray, J.N., Harrison, M.A. On the covering and reduction problems for context-free grammars. - J.ACM, 1972, 19,4,675-698.

3. Griffiths, M. LL(1) grammars and analysers. Compiler construction. Springer-Verlag, New York-Heidelberg-Berlin, 1976,57-83.

4. Graham, S.L. On bounded right context languages and grammars. - SIAM J. Comput. Vol. 3,3, 1974, 224-254.

5. Ахо А., Ульман Дж. Теория синтаксического анализа, перевода и компиляции. Том I. Синтаксический анализ. М., "Мир", 1978.

6. Ершов А.П., Покровский С.Б. Об унификации алгоритмических языков. "Проблемы кибернетики",

вып. 32. М., "Наука", 1977, с. 95-118.

7. Л е к а р м О. Практичность и переносимость системы построения трансляторов. -"Труды всесоюзного семинара по методам реализации новых алгоритмических языков", ч. I. Новосибирск, 1975, с. 47-73.

8. К н у т Д. О переводе (трансляции) языков слева направо. БКС. Языки и автоматы. М., "Мир", 1975, с. 9-41.

9. О л л о н г р е н А. Определение языков программирования интерпретирующими автоматами. М., "Мир", 1977.

10. Г а с е н а с А., П а л у о я Р. Транслятор BASIC-PL/1. -"Труды Таллинск. политехн. ин-та", 1978, № 439, с. 27-37.

11. В о о г л а й д А.О., Т о м б а к М.О. О проблемах редуцирования в грамматиках предшествования. -"Труды Таллинск. политехн. ин-та", 1975, № 386, с. 23-37.

12. В о о г л а й д А.О. Семантическое равенство распознавателей, работающих на грамматике LR(k) и грамматике предшествования с (1/1) ограниченным каноническим контекстом. - "Труды Таллинск. политехн. ин-та", 1976, № 411, с. 39-55.

13. В а д е р А.Р., В о о г л а й д А.О. Описание метаязыка в системе построения трансляторов. - "Труды Таллинск. политехн. ин-та", 1978, № 439, с. 83-92.

M. Lepp, A. Vooglaid

Practical Guidelines to Classical Methods of Syntactic Analysis

Summary

In this paper a new effective method of bottom-up syntactic analysis is presented. Some possibilities are shown to simplify semantic analysis using this method. The method is based on the three classical methods of syntactic analysis. They are - the simple precedence method, the method of (m,k)-bounded right-context, the method of mixed-strategy precedence. This method is effective to use on medium-sized computers.

№ 464

TALLINNA POLÜTEHNILISE INSTITUUDI TOIMETISED

ТРУДЫ ТАЛЛИНСКОГО ПОЛИТЕХНИЧЕСКОГО ИНСТИТУТА

УДК 519.24

Л.К.Выханду

ЭКСПРЕССМЕТОДЫ АНАЛИЗА ДАННЫХ

1. Введение. Как было указано в статье [1], классические методы анализа матрицы данных $X=(x_{ij})$, $i=1,\dots,N$; $j=1,\dots,M$ при помощи матрицы корреляций R и расстояний D трудоемки даже при работе на ЭВМ. Используя совпадение векторных пространств для матриц X, R, D, создали методику скоростного анализа данных, измеренных на интервальной шкале.

Возникает вопрос, нет ли аналогичной методики для данных, измеренных на номинальной или ординальной шкале. Оказывается, что простое обобщение понятия шкалы конформизма позволяет естественным образом использовать всю аппаратуру классического многомерного анализа, в том числе и скоростные методы анализа данных в случае интервальных шкал.

2. Шкала конформизма. Предположим, что собрана информация о множестве X, состоящем из N объектов. Пусть каждый объект $x_i \in X$ $(i=1,\dots,N)$ описывается кортежем из M дискретных случайных переменных $\langle x_{i1}, x_{i2}, \dots, x_{iM}\rangle$, где x_{ij} j-я переменная i-го объекта. Матрица $X=(x_{ij})$ $(i=1,\dots,N;\ j=1,\dots,M)$ есть матрица, описывающая множество X. Вектор-строка $r_i=(x_{i1}, x_{i2}, \dots, x_{iM})$ этой матрицы представляет множество значений случайных переменных i-го объекта, вектор-столбец $c_j=(x_{1j}, x_{2j}, \dots, x_{Nj})$ представляет множество значений j-й переменной по множеству объектов X.

Допустим, что каждая переменная x_{ij} -компонента вектора C имеет алфавит возможных значений $x_{jh} \in \mathcal{A}_j$, $h=1,\dots,k_j$. Эмпирическая частота события $x_{ij}=\alpha_{jh}$ в системе X есть сумма

$$n_j = \sum_{i=1}^{N} d_{ijh},$$

где
$$d_{ijh} \begin{cases} = 1, \text{ если } x_{ij} = \alpha_{jh} \\ = 0, \text{ если } x_{ij} \neq \alpha_{jh}. \end{cases}$$

На основании подобной эмпирической частоты можно получить различные оценки вероятности P_{jh} события $x_{ij} = \alpha_{jh}$. Например, в социологии обычно используют оценки

$$P_{jh} = n_{jh}/N.$$

В работе американских статистиков Bishop и др. [3] предложены т.н. квази-байэсовские оценки, которые стабильнее обычных и имеют меньший квадратный риск.

Сперва вычисляется вес

$$K = (1 - \sum ph^2)/(\sum ph^2 - 1/K_j)$$

и затем взвешенные оценки

$$P_{jh} = \frac{n_{jh}}{N+K} + \frac{K}{N+K} \cdot \frac{1}{K_j}.$$

Мы в своей работе будем придерживаться иной логики. Вначале матрица X представляется в частотном виде

$$Z = (Z_{ij}) \quad i = 1, \ldots, N, \; j = 1, \ldots, M,$$

где $Z_{ij} = n_{ij}$, т.е. частота события $x_{ij} = \alpha_{jh}$ в системе X.

Определение. Значением объекта на шкале конформизма K_i называется сумма частот n_{ij} значений x_{ij} объекта i.

Каков смысл такой замены и определения K_i.

Во-первых, сумма частот каждого объекта дает нам какую-то оценку его обычности - конформизма - в отношении всей группы объектов, легко обнаружить вероятного максимального или минимального конформиста в группе. Для максимального конформиста, который согласуется по каждому признаку с группой, надо просто найти максимальную частоту каждого признака, т.н. моду, и сложить эти моды. С другой стороны, сумма отличных от нуля минимальных частот от каждого признака, указывает нижний предел для объекта-отшельника, который всегда выбирает значения признаков, нетипичные для изучаемой группы.

Таким образом, из матрицы данных X на основе частотного преобразования получается новая матрица тех же размеров, которую уже можно обрабатывать как матрицу, заданную в интервальной шкале. Все признаки теперь соизмеримы. Соизмеримость признаков открывает новые возможности использования описанных ранее скоростных методик анализа данных. Для каждого объекта вычисляется сумма частот по его признакам. Этот процесс приводит системы объектов к т.н. шкале конформизма. Объекты с наибольшими значениями на этой шкале являются наиболее типичными для этой группы объектов; объекты с наименьшими значениями на этой шкале являются наименее типичными.

Шкала конформизма открывает многочисленные возможности упорядочения множества объектов по значениям этой шкалы. Одновременно упорядочение можно ввести и во множестве признаков. Для этого осуществляется построение гистограмм каждого признака P_j, а затем вычисляются суммы

$$H_j = \sum_{h=1}^{k} f_{hj}^2 ,$$

которые в обычном смысле служат измерителями вариабельности признаков, см. [2]. На основе вычисленных сумм H_j производится упорядочение признаков по величине возрастания (убывания) сумм H_j.

Таким образом, двустороннее упорядочение матрицы данных индуцирует перестановку в системе объектов и признаков. В результате проведения этой перестановки (которая на ЭВМ реализуется элементарно через ссылки, а не прямой перестановкой!) исследователь получает в свое распоряжение такое представление таблицы данных, которое говорит ему намного больше, чем начальное случайное расположение объектов и признаков. Отчетливо выявятся новые странности материала в случае, если какой-то объект имеет резко отличающее его от других объектов этой области значение (я) переменной(ых).

Пример. Пусть задана таблица данных X.

5	I	I	I	5	5	I	I
5	4	I	I	2	5	3	I
2	I	I	I	2	3	I	I
5	5	I	2	2	4	3	I
I	5	5	5	I	I	4	4
5	I	I	I	4	5	I	I
2	I	I	2	I	3	2	I
2	3	5	4	5	I	4	3
I	2	3	5	4	2	5	5
I	3	5	2	I	I	2	3

Составим гистограммы и вычислим $H_j = \sum_{h=1}^{k_j} f_{hj}^2$.

3	4	6	4	3	3	3	6
3	I	0	3	3	I	2	0
0	2	I	0	0	2	2	2
0	I	0	I	2	I	2	I
4	2	3	2	2	3	I	I
34	26	46	30	26	24	22	42

Заменим значения переменных в X на их частоты, получим Z в виде

4	4	6	4	2	3	3	6
4	I	6	4	3	3	2	6
3	4	6	4	3	2	3	6
4	2	6	3	3	I	2	6
3	2	3	2	3	3	2	I
4	4	6	4	2	3	3	6
3	4	6	3	3	2	2	6
3	2	3	I	2	3	2	2
3	I	I	2	2	I	I	I
3	2	3	3	3	3	2	2

Шкала конформизма приписывает объектам следующие значения и ранги:

Номер объекта	I	2	3	4	5	6	7	8	9	I0
Конформизм	32	29	3I	27	I9	32	29	I8	I2	2I
Ранг объекта	I	4	3	6	8	2	5	9	I0	7

После перестановки матриц данных X в порядке уменьшения конформизма значений и H_j получим

	Признаки								
объект	3	8	I	4	2	5	6	7	
I	I	I	5	5	I	5	5	I	32
6	I	I	5	I	I	4	5	I	32
3	I	I	2	I	I	2	3	I	3I
2	I	I	5	I	4	2	5	3	29
7	I	I	2	2	I	I	3	2	29
4	I	I	5	2	5	2	4	3	27
I0	5	3	I	2	3	I	I	2	2I
5	5	4	I	5	5	I	I	4	I9
8	5	3	2	4	3	5	I	4	I8
9	3	5	I	5	2	4	2	5	I2
	46	42	34	30	26	26	24	22	

Максимальный возможный конформизм для матрицы Z был бы 33, минимальный – II. Как видно из последней таблицы, первые места занимают сравнительно конформные объекты: все они близки к максимальному конформизму. Среди остальных объектов резко выделяется объект № 9, который только на один балл превышает полный минимум.

В заключение данного пункта рассмотрим требуемое количество операций для упорядочения объектов по шкале конформизма. Значение объекта x_i по шкале конформизма определяется как число

$$K_i = \sum_{j=1}^{M} n_{ij} ,$$

где n_{ij} эмпирическая частота осуществления события $x_{ij} = \alpha_{jh}$ в системе X. Необходимое количество вычислений для получения шкалы конформизма следующее: NM – число операций для получения гистограмм и ещё NM операций для вычисления K_i; всего $2NM$ операций, так что полное упорядочение объектов по шкале конформизма относится к быстрым методам в нашей классификации. После того,как шкала конформизма получена, можно поставить следующую задачу: сколько информации о всей системе объектов в целом получено (извлечено) из начальной неорганизованной матрицы данных X.

Насколько хороша шкала конформизма для такой оценки взаимного расположения объектов, выяснится в следующем пункте, где рассматриваются различные обобщения и уточнения шкалы конформизма.

3. Шкалы Адамара и Хаара для полного упорядочения объектов. Определение значения объекта по шкале конформизма как суммы его частот в заданной системе объектов кроме удовлетворительной содержательной трактовки предоставляет еще возможность эффективно использовать развитую ранее в [I] методику быстрых преобразований.

Допустим, что матрица данных представлена в частотном виде Z.

Легко показать, что после преобразования

$$\tilde{Z} = Z H,$$

где H преобразование Адамара (или Хаара), первый столбец матрицы $\tilde{Z}$ как столбец сумм строк Z совпадает со шкалой конформизма по системе объектов X. Следовательно, можно утверждать, что главная часть информации - вариабельности объектов в системе X, заключена в первом столбце матрицы $\tilde{Z}$. То обстоятельство, что шкала конформизма получается как часть преобразования Адамара (или Хаара), позволяет одновременно оценить "добротность" шкалы конформизма и произвести существенное улучшение этой шкалы.

В качестве измерителя степени важности первого столбца матрицы $\tilde{Z}$ предлагаются:

сумма $\tilde{Z}_j = \sum_{i=1}^{N} \tilde{z}_{ij}^2$ и отношение $\tilde{Z}_1 / \sum_{j=1}^{M} \tilde{Z}_j$.

Отношение $\tilde{Z}_1 / \sum_{j=1}^{M} \tilde{Z}_j$ показывает, какую долю вариабельности системы Z описывает шкала конформизма.

Матрица $\tilde{Z}$ представляет удобный путь обобщения шкалы конформизма сразу в двух направлениях. Во-первых, методом частичной ортогонализации [I] можно достичь большей вариабельности в проекции на главную ось системы Z (первый столбец $\tilde{Z}$). В этом смысле частичную ортогонализацию следует рассматривать как уточнение шкалы конформизма.

Во-вторых, осуществим переход на двумерное (визуальное) представление информации о взаимном расположении объектов системы X. Для получения приближенной картины расположения объектов системы X можно выбрать два столбца $\tilde{Z}$ с наибольшими суммами квадратов; выбранная пара компонент задает координаты объектов на плоскости. Эти координаты и определяют визуальную картину, которая довольно близка к картине, строящейся посредством двух первых собственных векторов матрицы расстояний D (отметим, что вычисление матрицы D требует $O(MN^2)$ операций плюс $O(N^3)$ операция для вычисления собственных векторов; вычисление же $\tilde{Z}$ - требует лишь $2NM+NM\log_2 M$ операций в случае преобразования Адамара или $2NM + 2NM$ для преобразования Хаара).

Наш опыт показывает, что для выявления структуры системы объектов X зачастую достаточно упрощенной визуальной картины, так как нередко удается выделить неоднородность системы объектов. Для неоднородных систем объектов (т.е. имеющих структуру) нет необходимости в столь точном определении собственных векторов, как для случая объектов с однородной структурой. В случае, если это уточнение все же требуется, в нашем распоряжении имеется два итерационных способа повышения точности: а) описанный ранее метод частичной ортогонализации [I]; б) сокращенный метод Варимакс, уточняющий лишь некоторые собственные векторы.

4. <u>Шкала влияния.</u> В данном пункте совершается переход в область для случая нелинейных обобщений. Нелинейные обобщения позволяют существенно обогатить арсенал средств эффективного анализа многопараметрических систем. Одним из возможных обобщений шкалы конформизма для таблиц данных может служить шкала Адамара. В этом разделе приводится еще один вариант, обобщающий шкалу конформизма.

Пусть задана матрица данных X и совершен переход к ее частотному виду Z. Определим меру вариации каждого объекта как сумму:

$$S_i = \sum_{j=1}^{M} Z_{ij}^2 .$$

Чем больше сумма S_i, тем конформнее объект i.

Исходя из меры вариации для каждого объекта, можно найти и меру вариации всей системы объектов как число

$$S = \sum_{i=1}^{N} S_i .$$

С одной стороны, ввиду аддитивности меры S имеет место равенство

$$S = \sum_{j=1}^{M} \sum_{i=1}^{N} Z_{ij}^2$$

и можно дать оценку доле вариабельности каждого признака в системе. С другой стороны, следующий естественный шаг приводит нас к понятию влияния объекта на вариабельность системы.

Определим влияние объекта как сумму квадратов, на которую уменьшается величина S, если из системы исключить этот объект, т.е. если ввести этот объект по каждому признаку в дополнительный класс неопределенности.

Так как для каждого признака j каждая частота h_j имеет еще $h_j - 1$ равных себе значений, то сумма квадратов частот для всей системы уменьшается на

$$g_{ij} = (h_{ij}-1)(h_{ij}^2 - (h_{ij}-1)^2) = 2h_{ij}^2 - 3h_{ij} + 1 . \qquad (\mathrm{I})$$

Суммируя по всем признакам, получим меру влияния объекта на систему в виде числа

$$g_i = \sum_{j=1}^{M} g_{ij} . \qquad (2)$$

Необходимые вычисления облегчаются, если применить таблицу

h	I	2	3	4	5	6	7	8	9	I0
g	0	3	I0	2I	36	55	78	I05	I36	I7I,

которая задается при помощи рекуррентной формулы

$$g(h+1)=g(h)+4h-1 .$$

На основании чисел g_{ij} вычисляются меры влияния, как числа

$$g_i=\sum_{j=1}^{M} g_{ij} ,$$

$$g_j=\sum g_{ij}$$

для объектов и признаков соответственно.

С тем, чтобы обнаружить связь между шкалой конформизма и шкалой влияния, раскроем выражение (2):

$$g_i = \sum_{j=1}^{M} g_{ij} = \sum_{j=1}^{M} (2h_{ij}^2 - 3h_{ij} + 1) =$$

$$= 2\sum_{j=1}^{M} h_{ij}^2 - 3\sum_{j=1}^{M} h_{ij} + M = 2S_i - 3K_i + M .$$

Таким образом, мера влияния объекта состоит из линейной комбинации его вариабельности и конформизма. Ввиду квадратичности шкалы влияния как функции она "усиливает" те объекты, которые конформнее. Метод шкалы влияния требует $O(MN)$ операций для составления гистограмм и $O(MN)$ операций для вычисления влияний объектов.

Ввиду аддитивности меры влияния g можно вновь воспользоваться уже примененной выше техникой уточнения и расширения шкалы конформизма.

Отправной точкой служит частотная матрица Z, каждый элемент которой z_{ij} заменяется на $g(z_{ij})$, и полученная матрица G обрабатывается по методу Адамара или Хаара.

5. Частотный метод частичного упорядочения объектов. В пункте 4 показано, что при помощи шкалы влияния за $O(NM)$ операций осуществимо полное упорядочение объектов. В этом пункте развивается другой подход, который дает больше информации о взаимных связях объектов. Этот подход основан на идее последовательного исключения наиболее слабых объектов из системы. Ясно, что исключение объекта из системы уменьшает и взаимовлияние оставшихся объектов ввиду уменьшения частот в системе.

В предыдущем пункте уже была выведена функция S_i измерения влияния объекта на систему:

$$S_i = \sum_{j=1}^{n} (3n_{ij} - 2n_{ij}^2 + 2),$$

где n_{ij} - частота значения x_{ij} в гистограмме j-го признака.

Исключение объекта k из системы меняет влияние каждого другого объекта i на величину

$$S_i(n_{ij}) - S_i(n_{ij}-1) = \sum_{j=1}^{M} \delta[3n_{ij} - 2n_{ij}^2 + 2 \cdot 3(n_{ij}-1) + 2(n_{ij}-1)^2 - 2] =$$
$$= \sum_{j=1}^{M} \delta(-4n_{ij} + 5),$$

где $\delta = I$, если $x_{kj} = x_{ij}$,

$\delta = 0$, если $x_{kj} \neq x_{ij}$,

т.е. суммирование происходит только по тем признакам объекта i, значение которых совпадает с соответствующими значениями исключаемого объекта k. Если у объектов k и i нет ни одного совпадающего значения, то естественно, что исключение объекта k на объект i никакого влияния не окажет.

При организации вычислительного процесса для каждого объекта i требуется запомнить, какой объект k своим исключением привел бы к наибольшему снижению влияния объекта i. Такая информация полезна как для группировки объектов, так и для превращения всей системы объекта в иерархическую систему.

Последовательное исключение объектов осуществимо при помощи следующего алгоритма, применяемого к матрице данных в частотном виде:

0I. Для $i = (1)N$ (найти $S_i = \sum_{j=1}^{M}(3n_{ij} - 2n_{ij}^2 + 2)$ и положить $P_i = 0$).

02. Найти $\max S_i$ и запомнить $k = i$.

03. N раз провести шаг 04.

04. Для всех не исключенных объектов $i (S_i \neq \infty)$ найти

$S_i = S_i + P_i$, где $P_i = 4T - 5L$, а T – сумма частот совпадающих значений признаков у объектов k и i, L – количество совпадений. Если P_i больше предыдущего P_i, то запомнить P_i вместе с индексом объекта k.

Во вспомогательной таблице частот W для всех исключенных значений объекта k вычесть единицу. Положить $S_k = \infty$.

Пример.

Пусть задана таблица данных

5	I	I	I	5	5	I	I
5	4	I	I	2	5	3	I
2	I	I	I	2	3	I	I
5	5	I	2	2	4	3	I
I	5	5	5	I	I	4	4
5	I	I	I	4	5	I	I
2	I	I	2	I	3	2	I
2	3	5	4	5	I	4	3
I	2	3	5	4	2	5	5
I	3	5	2	I	I	2	3

Матрица гистограмм W имеет вид (к каждой частоте прибавлена единица)

4	5	7	5	4	4	4	7
4	2	I	4	4	2	3	I
I	3	2	I	I	3	3	3
I	2	I	2	3	2	3	2
5	3	4	3	3	4	2	2

Влияния объектов приведены в нижеследующем ряду чисел по порядку:

–292, –259, –277, –233, –93, –292, –25I, –82, –32, –III.

Наиболее слабый объект 9. Процесс исключения этого и следующих за ним наиболее слабых объектов отражается в таблице:

–292	–259	– 277	–233	–93	–292	–25I	–82	–32	–III
–292	–259	–277	–233	–75	–285	–25I	–82	=	–I00
–292	–259	–277	–226	=	–285	–240	–53	=	–60
–285	–259	–266	–226	=	–285	–229	=	=	–32
–285	–259	–266	–2I5	=	–285	–204	=	=	=
–224	–2I3	–I9I	–I62	=	–224	=	=	=	=

-I7I	- I42	-I42	=	=	-I7I	=	=	=	=
-I04	=	-90	=	=	-I04	=	=	=	=
-49	=	=	=	=	-49	=	=	=	=
=	=	=	=	=	0	=	=	=	=

Из таблицы видно, что последовательность исключаемых объектов есть один из вариантов полного упорядочения объектов: 9, 5, 8, I0, 7, 4, 2, 3, I, 6. Сравнение этой последовательности с упорядочением по простой шкале влияния показывает, что имеются некоторые различия, которые в данный момент нас не интересуют.

6. Дерево влияния. Большой интерес для нас представляет то, что из этой же таблицы можно извлечь информацию для составления иерархического дерева максимального влияния системы объектов. Сравнение первой и второй строки таблицы показывает, что исключение объекта 9 дало наибольшее уменьшение влияния объекта 5 на систему (I8). Исключение объекта 5, в свою очередь, - объекта I0 (40), объекта 8 - объекта - I0 (28), объекта I0 - объекта 7(25), объекта 7 - объекта 3 (75), объекта 4 - объекта - 2 (7I), объекта 2 - объектов I и 6 (67), объекта 3 - объектов I и 6 (55), объекта I - объекта 6 (49).

На основе этих максимальных влияний, которые запоминаются в виде чисел P_i на шаге 04 алгоритма последовательного исключения объектов, получается дерево максимального влияния.

Пример. Начальные данные:

I	5	I	I	I	5	5	I	I
2	5	4	I	I	2	5	3	I
3	2	I	I	I	2	3	I	I
4	5	5	I	2	2	4	3	I
5	I	5	5	5	I	I	4	4
6	5	I	I	I	4	5	I	I
7	2	I	I	2	I	3	2	I
8	2	3	5	4	5	I	4	3
9	I	2	3	5	4	2	5	5
I0	I	3	5	2	I	I	2	3

Таблица частот W (к каждой частоте прибавлена единица для обеспечения монотонности схемы влияния):

I	4	5	7	5	4	4	4	7
2	4	2	I	4	4	2	3	I
3	I	3	2	I	I	3	3	3
4	I	2	I	2	3	2	3	2
5	5	3	4	3	3	4	2	2

Исключение объектов отражается так:

-292 -259 -277 -233 -93 -292 - 25I - 82 -32
- III = -I922

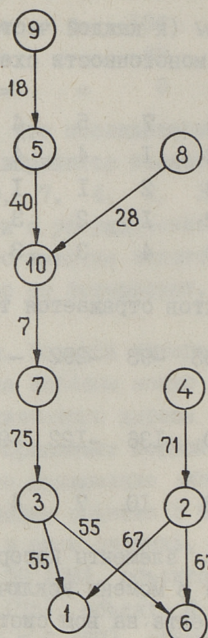
-240 -I60 -504 -200 -I36 -I22 -96 -464 = -I922

9 5 8 I0 7 4 2 3 I 6

Матрица $P(i,j)$, элементы которой показывают влияние объекта на другой в момент исключения (на главной диагонали влияние объекта на всю систему в момент исключения):

49	67	55	53	0	0	6I	7	0	0
0	I42	0	7I	0	0	46	0	0	0
0	52	90	49	0	0	75	II	0	0
0	0	0	I62	7	0	53	0	0	II
0	0	0	0	75	0	0	0	I8	0
49	67	55	53	0	0	6I	0	7	0
0	0	0	0	II	0	204	II	0	25
0	0	0	0	29	0	0	53	0	0
0	0	0	0	0	0	0	0	32	0
0	0	0	0	40	0	0	28	II	32

Если вывести на печать все значения P_i, вычисленные на шаге 04 алгоритма, то получится графическая картина, представляющая все связи, имеющие направленность в системе объектов.



Дерево максимального влияния.

7. Матрица расстояний по Хэммингу между объектами. Простое изменение алгоритма 0 (раздел 5) позволяет одновременно найти матрицу расстояний между объектами.

Как известно, расстояние Хэмминга [4] между двумя объектами определяется как

$$d(O_к,O_e) = \sum_{j=1}^{m} \delta(O_{кj}, O_{ej}),$$

где $\delta(O_{кj},O_{ej}) = I$, если $O_{кj} \neq O_{ej}$

0, если $O_{кj} = O_{ej}$.

В алгоритме 0 исключение какого-нибудь объекта из системы приводит к пересчету влияний оставшихся в системе объектов. Именно эти влияния и меняются на величину $4T - 5L$, где L – количество совпадений значений признаков у двух объектов $O_к$ и O_e. Отсюда сразу следует, что

$$d(O_к,O_e) = M - L.$$

Организация системы ссылок на матрице данных X позволяет эффективно реализовать процесс вычислений расстояний.

Пример. Для матрицы данных

5	I	I	I	5	5	I	I
5	4	I	I	2	5	3	I
2	I	I	I	2	3	I	I
5	5	I	2	2	4	3	I
I	5	5	5	I	I	4	4
5	I	I	I	4	5	I	I
2	I	I	2	I	3	2	I
2	3	5	4	5	I	4	3
I	2	3	5	4	2	5	5
I	3	5	2	I	I	2	3

Матрица расстояний D по Хэммингу имеет вид:

5	7	6	5	0	0	6	I	0	0
0	I4	0	7	0	0	5	0	0	0
0	5	9	5	0	0	8	I	0	0
0	0	0	I6	I	0	5	0	0	I
0	0	0	0	8	0	0	0	2	0
5	7	6	5	0	0	6	0	I	0
0	0	0	0	I	0	20	I	0	3
0	0	0	0	3	0	0	5	0	0
0	0	0	0	0	0	0	0	3	0
0	0	0	0	4	0	0	3	I	3

На главной диагонали стоят числа влияний (поделенные на I0) объектов на систему в момент исключения.

8. <u>Минимальное связывающее дерево для объектов и его модификации</u>. Указанная в предыдущем разделе возможность для вычисления матрицы расстояний обычно на практике себя не оправдывает, так как запоминание $N \times N$-матрицы требует слишком большого объема памяти. На практике значительно эффективнее при исключении каждого объекта i запомнить только тот объект, на который больше всего похож, т.е. тот объект h, для которого:

$$\max_{h} L(i,h),$$

h - индекс объектов, еще не исключенных из системы объектов. Эффективность данного приема гарантирована тем, что объекты исключаются из системы в порядке возрастания их влияния: для самых слабых объектов гарантируется поиск ближайших соседей.

Элементарен также переход к триангуляционному методу Ли, Слэйгла и Блума [5]. В качестве сопоставления предложенного метода минимального связывающего дерева объектов рассмотрим триангуляционный метод Ли, Слэйгла и Блума [5]. Эти авторы предложили метод представления системы объектов на плоскости следующим образом.

Вычисляется матрица расстояний D между объектами по Хэммингу. После этого выбирается один из объектов в качестве начала отсчета. Для всех объектов вычисляется расстояние до этого объекта, а затем расстояние до ближайшего соседа. Как показали Ли и др., этот метод позволяет вместо $N-1$ расстояний, корректно отображаемых методом минимального связывающего дерева, точно представить $2N-3$ расстояния и дать больше информации о структуре системы объектов X.

Предложенный в данном пункте метод эффективно реализует метод Ли и др. Объект, выбранный за начало отсчета, исключается первым, затем печатаются все его расстояния до других объектов и после этого продолжается процесс исключения по методу влияния с регистрацией наибольшего влияния каждого исключаемого объекта.

9. Стабильность полученного решения. Измеряемость влияния объектов системы по описанному в предыдущем подразделе способу представляет возможность для проверки стабильности полученных группировок объектов, а также выделенных подобластей матрицы данных. Для этого требуется при помощи генератора случайных чисел искусственно создать пропуски в матрице данных, а затем применить алгоритм последовательного исключения по методу влияний. Если при разных системах пропусков получаются одни и те же группы объектов или признаков, то эти группы устойчивы. Принципы получения статистических оценок с помощью генерации случайных пропусков в матрице данных излагаются в [4].

Литература

I. Выханду Л. Некоторые проблемы теории анализа данных. - "Тр. Таллинск. политехн. ин-та", № 366, 1974, с. 3-14.

2. А. Нильсон. Некоторые свойства сумм квадратов вероятностей и их математико-статистические приложения. Изв. АН ЭССР, серия техн. и физ.- матем. наук, XIУ, 1965, I, с. 79-93.

3. Bishop, Y. e.a. Discrete multivariate analysis. NY. 1975.

4. Hartigan, J. Clustering analysis. NY. 1975, 352 p.

5. Lee, R. e.a. A triangulation method for the sequential mapping of points from N-Space to two-Space. - IEEE Transactions on Computers, March 1977.

L. Vyhandu

Rapid Data Analysis Methods

Summary

In this article some new methods for sociological data analysis are described. To speed up multivariate description and analysis of data several principally new notions are introduced and used.

№ 464

TALLINNA POLÜTEHNILISE INSTITUUDI TOIMETISED

ТРУДЫ ТАЛЛИНСКОГО ПОЛИТЕХНИЧЕСКОГО ИНСТИТУТА

УДК 681.3.016:001.5

Ю.Г.Лааст-Лаас

ОБ ИНФОЛОГИЧЕСКОМ ПОДХОДЕ К БАЗАМ ДАННЫХ

Введение

Выработка и ведение интегрированных информационных систем требует эффективного сотрудничества между большим количеством людей разных специальностей и с разными навыками и уклонами образования. Пространство, в котором приходится вести исследования при создании автоматизированных информационных систем, слишком велико для упорядоченного обследования и слишком неизведанно для того, чтобы в нем могли разобраться люди, знания и опыт которых ограничивается рамками одной из существующих специальностей.

Важно, чтобы будущие пользователи системы могли предложить свои идеи на ранних стадиях разработки и участвовать в принятии принципиальных решений. Но нереально предполагать, что предметно ориентированные специалисты заинтересованы и способны сформулировать систему в терминах, свойственных обработке и организации данных в ЭВМ. С другой стороны, неразумно заставлять системщика, ориентированного на вычислительную технику, изучать частично покрывающиеся, но полностью не совместимые предметные модели в разных областях руководства и планирования, а также для установления целей информационной системы заставлять общаться со многими будущими пользователями, используя их разную специфически-профессиональную терминологию. Следовательно, нужен новый подход к созданию и ведению информационных систем, который дал бы назависимые от реализации средства описания информационно-логической структуры предметной области информационной системы и методы начального проектирования системы базы данных, а также непроцедурные средства представления информационных потребностей пользователя системы.

В связи с появлением мощных средств универсального характера (метасистем) для создания прикладных информационных систем тяжесть разработки названных прикладных систем переносится на базы данных (БД).

Весь комплекс научно-исследовательских работ, аналитической и творческой деятельности по изучению существующей системы и проектированию, разработке и внедрению системы БД разделим на два основных этапа: предварительное исследование и системирование.

В настоящей статье рассматривается один из возможных подходов к системированию баз данных - инфологический подход.

Предварительное исследование. Объектсистема

Каждая БД создается в интересах некоторой группы людей с целью дать им возможность систематически повышать эффективность выполнения своих должностных обязанностей. Эту группу людей будем называть субъектом системы или коллективным пользователем БД. Должностные обязанности субъекта всегда связаны с наблюдением, контролем за некоторой системой или управлением ею, называемой объектсистемой.

БД в инфологическом смысле является отображающей системой, т.е. системой, которая отражает объектсистему. Системирование БД может рассматриваться как отображение объектсистемы в БД. Из-за стоимостно-временных ограничений отражение будет всегда аппроксимацией реальной системы. Для выполнения своих обязанностей субъекту нужна только определённая часть информации. Правильное ограничение БД "по ширине и глубине" изображения объектсистемы - определение области определенности БД - является предпосылкой ожидаемой эффективности системы.

Каждый пользователь БД воспринимает объектсистему в двух аспектах: компетентности и своих должностных прав и обязанностей. Оставим вопрос как именно формируется предметная область (S_i) отдельного пользователя открытым и определим S_i как некоторую подсистему объектсистемы, представляющую интерес для пользователя.

Уточним понятие субъекта: $u = \{u_1, u_2, \ldots, u_n\}$, где u_i $(i=1,\ldots,n)$ – группа отдельных пользователей БД, которые имеют тождественную предметную область S_i. Для краткости изложения ниже u_i будем называть пользователем.

Отметим, что в процессе системирования БД S_i до некоторой степени меняется, так как изменяются цели функционирования объектсистемы, должностные обязанности u_i и накапливается опыт использования возможностей БД. Определение S_i требует в каждом конкретном случае тщательных предварительных исследований со стороны специалистов по управлению и возможных реорганизаций в системе управления объектсистемой (диагностика ее системы управления, выяснение тенденций развития объектсистемы, синтез задач управления, постановка целей, разработка мероприятий для достижения целей, в том числе составление должностных руководств).

Системирование. Инфологические модели

Системирование БД состоит из ряда последовательных этапов: – моделирование объектсистемы, т.е. выделение некоторой ее подсистемы – области определенности БД – и знаковое отображение ее в предметных моделях;

– последовательно-ступенчатый переход от предметных моделей (UM) к промежуточной инфологической модели (IM), от IM к модели структур данных (DM) и от DM к конечной модели, которой является БД.

В настоящей статье под моделью подразумевается некоторая абстракция, представление реальности. Инфологические модели являются как определенными ступенями познания и описания объектсистемы, отображениями объектсистемы в определенных языках, так и ступенями инфологического описания БД. Каждая инфологическая модель соответствует определенному аспекту описания объектсистемы и БД:

UM – аспект пользователя;

IM – инфологический аспект;

DM – аспект структур данных;

БД – аспект реализации.

Для последовательного ступенчатого описания функциональных характеристик БД – от сравнительно абстрактных требований пользователей и общих целей системы к конкретной реализации – инфологические модели дополняются соответственными описаниями БД, рассмотрение последних не умещается в рамки данной статьи.

Каждый пользователь U_i имеет определенную предметную область S_i, для описания которой требуется соответственная предметная модель UM_i. Так как каждая модель UM_i соответствует определенному полю представлений отдельной группы специалистов, она может быть неполной с точки зрения субъекта системы и отдельно взятая не может служить исходной моделью системирования. Выработка предметной модели под силу лишь людям, непосредственно участвующим в практической деятельности управления данной объектсистемой. В UM для описания структуры, состава и функционирования объектсистемы применяются понятия, семантика которых может быть выражена только в узко специализированной терминологии. UM не требует строгой формализованности, она может быть представлена в самой разной терминологии, средствами естественного языка, графическими изображениями, символикой различных областей науки.

В следующем разделе рассматриваются общие инфологические понятия, связанные с моделированием и предметной моделью.

Для сглаживания разногласий в отображении объектсистемы и требованиях, предъявляемых к системе БД различными пользователями, а также с целью сопоставления и консолидации разных точек зрения, предметные модели преобразовываются в единую промежуточную инфологическую модель. Этот этап называется спецификацией. На этапе спецификации производится анализ разных предметных моделей, определяется элементарный уровень описания объектсистемы. Посредством минимальных информационных структур –

элементарных сообщений[1] - описывается информационная база БД с однородной структурой. IM является инфологической основой БД, теоретико-множественной интерпретацией информационной базы. Эта модель лишь промежуточная и является результатом анализа. В IM даны ориентиры для проектирования БД: сформулированы информационные потребности субъекта, даны инфологические структуры данных, представлены требования ко внутренним и внешним способностям БД (см. [7]).

Представлению IM последует синтез структур хранения и обработки данных, определение методов и путей доступа к данным и всех остальных характеристик БД, связанных с конкретной реализацией. Этот этап называем проектированием БД. В инфологическом смысле проектирование является преобразованием IM в модель структур данных (DM).

Последнее преобразование модели DM → БД называется реализацией, но в данной статье этот этап системирования не рассматривается.

Моделирование объектсистемы. Предметная модель

Моделирование проводится специалистами предметной ориентации, т.е. специалистами по управлению объектсистемой, будущими пользователями БД. Результатами этого этапа являются предметные модели разных пользователей.

Этап моделирования рассматривается нами как состоящий из трех стадий: восприятие объектсистемы, индукция и абстрагирование. Названные стадии находятся в логической последовательности, но это не значит, что процесс моделирования обязательно должен быть линейным. Получаемые на каждой из стадий моделирования результаты не окончательны. Так как процесс познания сменяется процессом реализации познанного, результаты постоянно пересматриваются

---

[1] Подобные "высказывания об элементарном явлении действительности, которые удовлетворяют требованиям универсальности, минимальности, полноты, явной выраженности и формализованности", называют в монографии [I] отображающими элементами.

и уточняются. Дело в том, что восприятие перемежается индукцией и абстрагированием - моделированию присуща цикличность, итеративность. Бесполезны попытки найти решения задач моделирования линейным путем, в лучшем случае методика моделирования может служить лишь уменьшению цикличности и увеличению линейности процесса. При выборе методики и конкретных способов фиксирования промежуточных результатов и представления их в моделях важно учитывать необходимость внесения многократных поправок и модификации.

Восприятие объектсистемы

Исходим из аксиомы, что предметная область состоит только из объектов, связанных между собой определенным образом. Интуитивно объект есть нечто реальное или абстрактное, отличимое и значительное в предметной области (или определенная часть этого), в чем заинтересован субъект БД. Объект является фундаментальным инфологическим понятием, которое формально не определяется. Однако конкретный объект может быть определен формально. Объект, который формально определим через другие объекты, называется составным, если же он формально не определим, его называют элементарным. Множество объектов обозначаем через E.

В функциональной и логической связанности, во взаимодействии определенных объектов возникают взаимосвязи. Взаимосвязью называем зависимость друг от друга двух или более объектов, существенную по мнению субъекта. Взаимосвязи объединяют объекты в единое целое - в систему. Множество взаимосвязей обозначаем через C. Количество объектов, которые соединяются взаимосвязью $c \in C$, назовем разрядом взаимосвязи.

В стадии восприятия выбираются объекты ($E' \subseteq E$) и взаимосвязи ($C' \subseteq C$), представляемые в моделях. Выбор E' и C' определяется информационными требованиями субъекта.

Индукция

Исследуя подробнее отдельные объекты в их взаимосвязях, можно выделить отдельные свойства объектов. В свойствах проявляется и специфичность и общность объектов. Под свойством понимается некоторая существенная на взгляд субъекта и интересующая его качественная или количественная характерная черта объекта. Свойства не могут рассматриваться и не имеют смысла отдельно от объекта и их невозможно сравнивать по степени важности по отношению к нему. Обозначим множество свойств всех объектов множества E через P. Одним из результатов индукции является выбор P' (причем $P' \subseteq P$)-множества свойств, которые находят представление в предметной модели.

При исследовании объектсистемы выявляется ее динамичность. Объекты возникают и исчезают, изменяются их свойства и взаимосвязи. При изучении динамических явлений в объектсистеме важное значение приобретает понятие времени. Объекты, их взаимосвязи и свойства существуют и развиваются во времени. Надо определить временное расположение объектов, взаимосвязей и свойств (ответ на вопрос "когда?") и длительность их существования. Пусть существование каждого объекта, каждого его свойства и каждой взаимосвязи характеризуется некоторым интервалом времени t. Обозначим множество всех рассматриваемых интервалов времени через T.

Теоретико-множественное представление предметной области следующее: $S = \langle E, C, P, T \rangle$, а области определения предметных моделей следующее: $S' = \langle E', C', P', T' \rangle$. При этом каждая отдельная предметная модель UM_i имеет область определения:

$$S'_i = \langle E'_i, C'_i, P'_i, T'_i \rangle ; S' = \bigcup_{i=1}^{n} S'_i, \quad E' = \bigcup_{i=1}^{n} E'_i, \quad C' = \bigcup_{i=1}^{n} C'_i,$$

$P' = \bigcup_{i=1}^{n} P'_i$ и $T' = \bigcup_{i=1}^{n} T'_i$, где n - количество предметных моделей.

Стадии восприятия и индукции тесно связаны со стадией абстрагирования, где практически фиксируются все результаты наблюдения и исследования объектсистемы - строится предметная модель.

Абстрагирование

На этом этапе создается знаковое отображение объектсистемы - предметная модель. Отображение объектсистемы можно вести двумя способами [I]: прямым, через непосредственное наблюдение, и косвенным, через формальный анализ документооборота. Эти способы часто применяются комбинированно. Однако при этом очень важно не опустить основную цель - получить отображение объектсистемы, соответствующее интересам и требованиям субъекта, и не сводить ее к формальному копированию существующего отображения, которое этим требованиям может не соответствовать (быть слишком детальным, односторонним, устаревшим и т.п.).

Приведем в определенное соответствие понятия объектсистемы и отображающие их понятия модели.

Объекту $e \in E_i'$ соответствует в UM_i представление объекта - коротко: компонент $k \in K_i$ (K_i - множество компонентов модели UM_i). Фундаментальным будем называть компонент, соответствующий элементарному объекту, производным - компонент, соответствующий составному объекту.

Взаимосвязи $c \in C_i'$ соответствует в UM_i представление взаимосвязи - коротко: отношение $r \in R_i$ (R_i - множество отношений модели UM_i). Отношения между компонентами фиксируют взаимосвязи между объектами.

Свойству $p \in P_i'$ соответствует в UM_i представление свойства - коротко: атрибут $a \in A_i$ (A_i -множество атрибутов модели UM_i).

Интервалу времени $t \in T_i'$ соответствует в UM_i представление интервала времени - коротко: отрезок времени $z \in Z_i$ (Z_i - множество отрезков времени модели UM_i).

Определим отображение $\mu : S_i \to UM_i$ так, что

$\mu : P_i' \to A_i$, $\mu : C_i' \to R_i$, $\mu : E_i' \to K_i$ и $\mu : T_i' \to Z_i$.[I]

[I] Для удобства изложения здесь и ниже, где нет необходимости подчеркивать различия отображений (μ, n, ν, γ, β), в абстрагировании разных UM_i будем опускать индекс i при обозначении отображений.

Составные единицы моделей UM будем строить из слов в алфавите L. Через L^* обозначим определенный язык модели — множество непустых слов над алфавитом L.

Пусть заданы два отображения:

$$n : (E'_i \cup C'_i \cup P'_i) \to L^*,$$
$$\nu : P'_i \to L^*.$$

На P'_i определим комбинацию n и ν:

$$n\nu : P'_i \to L^* \times L^*.$$

Отображение $n\nu$ каждому из $p \in P'_i$ ставит в соответствие пару $(n(p), \nu(p))$ элементов из L^*.

Будем называть

$n_к = n(e)$ – именем компонента $к = \mu(e)$,

$n_r = n(c)$ – именем отношения $r = \mu(c)$,

$n_a = n(p)$ – именем атрибута $a = \mu(p)$,

$\nu_a = \nu(p)$ – значением атрибута $a = \mu(p)$.

Замечание I. Ограничим отображение n так, чтобы все имена были бы уникальны в пределах данной UM_i.

Замечание 2. "Неизвестно" является возможным значением атрибута, значению "неизвестно" соответствует резервированное слово, обозначаемое через $\lambda (\lambda \in L^*)$.

Формально атрибутом $a = \mu(r)$ назовем пару $\langle n_a, \nu_a \rangle$, где $n_a = n(p)$ и $\nu_a = \nu(p)$.

Пусть заданы два отображения: $\gamma : E'_i \to P'_i$ и $\beta : E'_i \to C'_i$, γ – является отображением, которое каждому объекту ставит в соответствие множество его свойств ($\gamma(e) = \{P_e | e \in E'_i, P_e \subseteq P'_i\}$); β – является отображением, которое каждому объекту ставит в соответствие множество взаимосвязей, связывающих его с другими объектами ($\beta(e) = \{C_e | e \in E'_i, C_e \subseteq C'_i\}$).

Формально компонентом $к = \mu(e)$ назовем пару $\langle n_к, A^к \rangle$, где $n_к = n(e)$, $A^к = \{n\nu(p) | p \in \gamma(e)\}$ ($A^к$ – множество атрибутов компонента $к$).

Формально m– местным отношением $r = \mu(c)$ (c – взаимосвязь с разрядом m) назовем пару $\langle n_r, K_r \rangle$, где $n_r = n(c)$, K_r – упорядоченное множество:

$$K_r=\{(n_{k1},\ldots,n_{km})\,|\,n_{k_i}=n(e_i),\quad c\in\beta(e_i)\quad (i=1,\ldots,m)\}.$$

Пусть даны два отображения: $\tau_1: T\to L^*$ и $\tau_2: T\to L^*$. Назовем $\tau_1(t)$ начальным моментом отрезка времени $z=\mu(t)$ и $\tau_2(t)$ конечным моментом отрезка времени $z=\mu(t)$.

Формально отрезком времени $z=\mu(t)$ назовем пару $\langle\tau_{1z},\tau_{2z}\rangle$, где $\tau_{1z}=\tau_1(t)$ и $\tau_{2z}=\tau_2(t)$.

Образом компонента $\langle n_k, A^k\rangle$ $(k=\mu(e))$ назовем множество $I_k=\{n_a|n_a=n(p),\ p\in\gamma(e)\}$. Компоненты, и образы и семантика которых являются тождественными,[I] назовем однородными. Множество однородных компонентов назовем видом компонентов (B_k).

Обозначим семейство видов через B.

Расширим отображение n на B: $n: B\to L^*$.

Будем $n(B_k)$ $(B_k\in B)$ называть именем вида компонентов.

Образом отношения $\langle n_r, K_r\rangle$ $(K_r=(n_k,\ldots,n_{k_m}))$ назовем множество $I_r=\{(n(B_{k_1}),\ldots,n(B_{k_m}))\,|\,k_1\in B_{k_1},\ldots,k_m\in B_{k_m}\}$. Отношения, и образы и семантика которых являются тождественными, назовем однородными. Множество однородных отношений назовем видом отношений (B_r).

Будем $n(B_r)$ $(B_r\in B)$ называть именем вида отношений. Имена видов компонентов/отношений являются обобщенными именами их элементов. Образом вида называем образ любого из его элементов, т.е. $I_{B_k}=I_k$ $(k\in B_k)$ и $I_{B_r}=I_r$ $(r\in B_r)$.

Формально виды B_k представляются парой $\langle n_{B_k}, I_{B_k}\rangle$, где $n_{B_k}=n(B_k)$ и виды B_r – парой $\langle n_{B_r}, I_{B_r}\rangle$, где $n_{B_r}=n(B_r)$.

Условно видом атрибутов назовем множество атрибутов, имеющих тождественную семантику.

Для структуризации и описания предметной модели целесообразно осуществить более обобщенное классифицирование представлений объектов, взаимосвязей и свойств.

Определяем класс компонентов (отношений, атрибутов) как совокупность компонентов (отношений, атрибутов), каж-

---

I Вопросы тождественности семантики могут решиться только экспертами соответствующей предметной ориентации.

дый из которых удовлетворяет определенному аспекту классифицирования. Аспектом классифицирования называется точка зрения, связанная с определенными интересами субъекта, на основе которых определяется общность объектов (взаимосвязей, свойств). Классификатором называется совокупность правил классифицирования. Классификацией назовем систему классов, определяемую классификатором.

Классификация определяет частичную упорядоченность компонентов, отношений и атрибутов, что способствует систематизированию работ системирования.

Обозначаем множество классов применяемой классификации через Y. Расширяем отображение n на множество Y:

$n : Y \to L^*$. Называем $n(y)$ именем класса $y \in Y$.

Так как понятие класса обширнее понятия вида, то $B \subset Y$.

Соответственно интересам разных групп будущих пользователей возможно составить ряд различных классификаторов. Классификация является результатом умственных процессов (индукции и абстрагирования) по обобщению с организации представления реального мира. Классы рассматриваются как абстракции и таким образом не являются структурами, непосредственно связанными с объектсистемой и со стадией восприятия. Классы являются всегда более субъективными составными модели , отражающими определенный аспект рассмотрения объектсистемы. В основе иерархической классификации всегда лежит некоторое упорядоченное рассмотрение отдельных характеристик, которое всегда до некоторой степени произвольно.

Формально класс представляется парой $\langle n_y, I_y \rangle$, где $n_y = n(y)$ - имя класса и I_y - образ класса (аспект классифицирования), который в свою очередь формально определим для каждого конкретного класса.

Результатом моделирования объектсистемы является ряд предметных моделей разных пользователей. Формально предметная модель UM_i дана пятеркой $\langle L^*, K_i, R_i, A_i, Z_i \rangle$, где

L^* - язык модели,

K_i - множество компонентов модели $\langle n_k, A^k \rangle$,

R_i - множество отношений $\langle n_r, K_r \rangle$,

A_i – множество атрибутов $\langle n_a, v_a \rangle$,

Z_i – множество отрезков времени $\langle \tau_{1z}, \tau_{2z} \rangle$.

Описание модели UM_i дается через классификацию компонентов, отношений и атрибутов, при этом особо выделяют виды. Каждый из классов $y = Y_i$ описывается посредством его образа и семантики.

Спецификация. Промежуточная инфологическая модель

Спецификация является вторым этапом системирования. Он включает в себя анализ описаний предметных моделей UM_i, возможное инфологическое соединение UM_i, их информационное отображение и создание единой промежуточной инфологической модели.

Пусть даны $UM_1, \ldots, UM_n$ с частично покрывающимися соответственными областями определения $S'_1, \ldots, S'_n$. Определим область определения IM как $S' = \bigcup_{i=1}^{n} S'_i$.

Из-за специфики интересов разных пользователей и субъективности отображения предметной области появляются разногласия между моделями UM_i и UM_j в областях $S'_i \cap S'_j$ $(i,j = 1,\ldots,n;\ i \neq j)$. На основе анализа предметных моделей можно выделить некоторые типы разногласий.

1. Наблюдается разница в именах семантически тождественных, а также сходство имен семантически разных компонентов, отношений, атрибутов, видов, классов, т.е. разница в отображении n.

2. Наблюдается разница в образах семантически тождественных компонентов, отношений и соответствующих видов, т.е. разница в отображениях β и γ.

3. Наблюдается разница в образах семантически тождественных классов, т.е. в аспектах классифицирования.

4. Наблюдается разногласие в рассмотрении явлений объектсистемы. Явление представляющее специальный интерес для одного из пользователей, может рассматриваться им как объект, в то время, как это же явление другим пользовате-

лем может рассматриваться лишь свойством некоторого другого объекта или взаимосвязью между некоторыми объектами.

Если разногласия типов I–3, возникшие на стадиях индукции и абстрагирования, сглаживаются сравнительно легко с помощью компетентных экспертов и автоматизированных средств моделирования, то справиться с разногласиями типа 4, возникшими еще на стадии восприятия, удается далеко не всегда. Если пользователи не могут прийти к общему мнению в рассмотрении объектсистемы, что делает создание единой инфологической модели невозможным, то целесообразно создание нескольких баз данных. Уменьшить вероятность возникновения разногласий последнего типа можно за счет умелого создания методики моделирования, снабженной конкретными рекомендациями для инфологического описания предметной области, выработанными на основе накопленного опыта.

Разногласия типов I и 2 могут сглаживаться операциями частичного переименования элементов моделей (в том числе возможно построение единой структуры составных имен и т.п.) и дополнения образов видов. Дополнение образа вида компонентов B_k приводит к увеличению количества связанных с компонентами $k \in B_k$ атрибутов. Теоретически возможна такая операция и с образом вида отношений B_r. Если r — m-местное отношение ($r \in B_r$), то операция дополнения образа вида B_r приведет к росту величины m. Однако практически целесообразность операций дополнения видов отношений сомнительна, так как невероятно, что семантика m-местного и $(m+k)$-местного ($k=1\ldots$) отношения может считаться тождественной.

Для выделения дополнительных типов разногласий введем понятия фундаментального и производного отношения/атрибута. На базе значений некоторых атрибутов компонента $k \in K_i$ с помощью математико-логических операций можно сгенерировать множество значений новых вычисленных составных атрибутов компонента $k \in K_i$. Обозначим множество атрибутов компонента k через A^k и множество правил генерации значений составных атрибутов компонента k через $G_{A^k}=\{g_1,\ldots,g_m\}$. Если $g \in G_{A^k}, (a_1,\ldots,a_n) \subset A^k$ и $g(v_{a_1},\ldots,v_{a_n}) = v_{a_{n+1}}$, то $a_{n+1} \in A^k$. Базой A^k назовем такое подмножество $A^k_G \subset A^k$, которое минимально в том смысле, что не найдется такого $a \in A_G^k$, при исключении которого из A_G^k удалось бы с помощью G_{A^k} сгенерировать все $V_a(a \in A^k)$. Атрибуты $a \in A_G^k$ называются фундаментальными, атрибуты $a \in (A^k \setminus A_G^k)$ – производными. Множество $G_{A^k} = \{g_1, \ldots, g_m\}$ называется генератором множества A^k.

Отношения могут генерироваться аналогично атрибутам. Обозначим множество отношений, связывающих компоненту k с другими компонентами модели, через R^k и множество правил генерации отношений – $G_{R^k} = \{g_1, \ldots, g_m\}$. Если $(r_1, \ldots, r_m) \subset R^k$, $g \in G_{R^k}$ и $g(r_1, \ldots, r_n) = r_{n+1}$, то $r_{n+1} \subset R^k$. Базой R^k назовем подмножество $R_G^k \subset R^k$, которое минимально в том смысле, что не найдется такого $r \in R_G^k$, при исключении которого из R_G^k удалось бы с помощью G_{R^k} сгенерировать R^k. Отношения $r \in R_G^k$ называются фундаментальными, отношения $r \in (R^k \setminus R_G^k)$ – производными.

При необходимости ограничения все m-местные отношения могут рассматриваться производными на базе бинарных $(m = 2)$ отношений.

Назовем компоненты, атрибуты и отношения элементами модели. Внутримодельными понятиями назовем имена элементов модели, имена классов, классификаторы и значения атрибутов.

Производные элементы модели формально определимы через внутримодельные понятия. Фундаментальные элементы модели могут определяться только неформально, т.е. внемодельными понятиями. Далее продолжим перечень типов разногласий.

5. Наблюдается разногласие в том, что элемент модели может считаться фундаментальным в пределах модели UM_i, а в пределах UM_j семантически тождественный элемент может считаться производным.

6. Наблюдается разногласие в том, что в целом семантически разные элементы, считаемые фундаментальными в двух предметных моделях, могут иметь общую и тождественную в семантическом смысле часть.

Для устранения разногласий типа 5 и 6 в ходе преобразования предметных моделей в единую IM, требуется элементарный уровень описания области определения (S').

Элементы модели, которые хоть в одной из UM рассматриваются производными, аналогично должны рассматриваться и в модели IM. В частично покрывающихся по семантике фундаментальных элементах модели в двух различных UM, выясняется наиболее крупная общая единица, называемая частью, которая в семантическом смысле одинакова для обоих элементов. Эти общие семантические части являются основой для фиксирования новых, более элементарных фундаментальных элементов (см. пример).

Пример.

ИМЕНА ВИДОВ ФУНДАМЕНТАЛЬНЫХ ЭЛЕМЕНТОВ В МОДЕЛЯХ

	UM_1	UM_2	IM
виды компонентов	"строка документа Е/32"	"колонка документа Е/32"	"показатель на документе Е/32"
виды атрибутов	"длина диагонали квадрата"	"площадь квадрата"	"длина диагонали квадрата" [I]
виды отношений	"операция соединения (деталь,узел)"	"операция сварки (деталь, узел)"	I. "операция сварки (деталь,узел)" 2. "операция монтажа (деталь,узел)"

I Здесь в принципе не важно, который из видов считать видом фундаментальных атрибутов, однако выбор "длина стороны квадрата" был бы явно нецелесообразным, так как вместо одного производного атрибута появилось бы два.

При делении компонентов возникают новые виды фундаментальных компонентов, это сопровождается появлением новых видов отношений. При этом некоторые бывшие фундаментальные отношения могут оказаться производными на базе новых отношений.

Информационное отображение модели требует введения понятия информационной единицы. Инфологическую единицу информации назовем сообщением. Сообщения соответствуют определенным комбинациям элементов предметной модели – ситуациям. Сообщения представляются через наименования элементов предметной модели. Наименование является логическим символом элемента UM, но не совпадает с ним. Наименование может иметь ряд различных представлений. Наименование может даваться явно – через имя именуемого элемента, или косвенно – комбинацией имен других элементов. Наименование может быть однозначным, которое локализирует определенный элемент модели, или многозначным, которому можно поставить в соответствие несколько элементов.

Базовые информационные единицы называются элементарными сообщениями. Если $\langle x, y, z \rangle$ является выражением, где x m-местное упорядоченное множество явных однозначных наименований компонентов, y – явное и однозначное наименование атрибута $a \in A_G^k$ или отношения $r \in R_G^k$ и z – отрезок времени $z \in Z$, то $\langle x, y, z \rangle$ называется полным элементарным сообщением (коротко: э-сообщением).

Э-сообщения подразделяются на два типа: атрибута и отношения. Э-сообщение типа атрибута имеет вид $\langle N_k, N_a, z \rangle$, где $N_k = n_k$, $N_a = (n_a, \nu_a)$, $k = \mu(e)$, $n_a = n(p)$, $\nu_a = \nu(p)$, $z = (\tau_1(t), \tau_2(t))$ и оно несет информацию о том, что объект e в интервал времени t имеет свойство p. Э-сообщение типа отношения имеет вид $\langle (N_{k_1} \dots N_{k_m}), N_r, z \rangle$,

где $N_{k_i} = n_{k_i}$, $k_i = \mu(e_i)$, $N_r = n_r$, $r = \mu(c)$, $z = (\tau_1(t), \tau_2(t))$

и несет информацию о том, что в течение интервала времени t объекты $e_1 \dots e_m$ находятся во взаимосвязи c.

Любое сообщение может быть представлено комбинацией Э-сообщений. Формализованность при этом обеспечивается жесткой структурой э-сообщений. Возможная методика агрегирования составных сообщений и представления основных типов информационных запросов на базе совокупности Э-сообщений представлена в монографии [7].

Неполные э-сообщения соответствуют определенным множествам ситуаций. Информационные запросы могут рассматриваться как неполные э-сообщения или как комбинации э-сообщений, часть из которых является неполной.

Множество информационных единиц с однородной структурой - э-сообщений - является теоретико-множественной интерпретацией информационной базы БД, определяет ее инфологический объем.

В целях создания возможности общения с базой данных для каждого пользователя разрабатываются средства, позволяющие представлять информационные потребности в инфологической, непроцедурной форме.

Информация, связанная с определенным объектом e, локализируется по наименованию компонента $k = \mu(e)$. Наименование будет явным, если k фундаментальный компонент или косвенным, если k производный компонент. Косвенное наименование производного компонента дается через имена определенных фундаментальных компонентов и связывающих их отношений.

Информация, относящаяся к некоторой взаимосвязи c (к некоторому свойству p), локализируется по наименованию соответствующего отношения $r = \mu(c)$ (атрибута $a = \mu(p)$). Наименование будет явным, если r - фундаментальное отношение (a - фундаментальный атрибут), или косвенным, если r - производное отношение (a - производный атрибут). Косвенное наименование дается через имена определенных фундаментальных отношений $(r_1, \ldots, r_n)$ (атрибутов $(a_1, \ldots, a_n)$) и правила генерации отношения $r = g(r_1, \ldots, r_n)$ (значения атрибута $v_a = g(v_{a1}, v_{a2}, \ldots, v_{an})$).

Проектирование. Модель структур данных

Проектирование является третьим этапом системирования, который полностью проводится специалистами-системщиками, ориентированными на организацию и обработку данных в ЭВМ. На этом этапе производится синтез информационных структур хранения и обработки на базе совокупности э-сообщений. В инфологическом смысле проектирование рассматривается как преобразование IM в модель структур данных.

Поиск эффективных способов составления информационных структур обосновывается известной теоремой информационной теории [8]:

$$E(S) \geqslant H(x) = -\sum_{i=1}^{n} p_i \log_2 p_i ,$$

где $E(S)$ – среднее количество шагов поиска при любой системе идентификации элементов множества $x = \{x_1, \ldots, x_n\}$, вероятности появления элементов – $p_i (i=1 \ldots n)$, $\sum_{i=1}^{n} p_i = 1$, $H(x)$ – энтропия множества X.

Исходными данными проектирования служат классификации элементов моделей UM, образы классов, описания основных видов информационных запросов и т.п.

При генерации структур приходится сталкиваться с проблемами выбора тех э-сообщений, которые целесообразно соединять в одну единицу хранения, и тех э-сообщений, которые целесообразно дублировать и т.д.

Для нахождения более эффективных вариантов генерации единиц хранения разумно пользоваться автоматизированными средствами проектирования.

В модели структур данных дается описание единиц хранения и обработки данных, определение методов и процедур доступа к данным и других характеристик, связанных с конкретной реализацией БД.

Литература

1. Зайцев Н.Г. Принципы информационного обеспечения в системах переработки информации и управления. Киев, "Наукова думка", 1976.

2. Михновский С.Д. Описание внешнего пространства данных автоматизированных информационных систем. "УСиМ", 1976, № 4.

3. Михновский С.Д. Некоторые вопросы модельного описания информационно-поискового процесса в базе данных. – В сб. "Вопросы проектирования банков данных". Киев, ИК, 1977.

4. Ерема-Еременко А.А., Михновский С.Д. Использование некоторых процедур семантического анализа описания исходных данных при проектировании

автоматизированных информационно-справочных систем. - В сб. "Вопросы проектирования банков данных". Киев, ИК, 1977.

5. Черняк Ю.И. Системный анализ в управлении экономикой. М., "Экономика", 1975.

6. Цуников Ю.В., Челноков Н.И., Зубов В.С. Принципы построения информационной базы АСУ на ЭВМ III поколения. М., МЭИ, 1977.

7. Sundgren, B. An infological approach to data bases. Stockholm, 1973.

8. Dabrowski, A. O teorii informacji. Warszava, 1974.

9. Langefors, B. Information systems. - Information Processing 74. North-Holland Publ. Co., 1974.

10. Moulin, P., Randon, J. etc. Conceptual model as a data base design tool. - Modelling in Data Base Management Systems, North-Holland Publ. Co., 1976.

J. Laast-Laas

On Infological Approach to Data Bases

Summary

Urgent need for data base systemeering means which should support coordinated interaction and mutual understanding between subject-matter-oriented specialists and computer-oriented systemeers is pointed out.

Infological approach to data bases, basic concepts of infological theory and infological intermediary models are introduced.

Object-system - data base mapping as a set of homomorphic submappings (object-system → subject-matter-model → infological model → datalogical model → data base) is described.

автоматизированных информационно-справочных систем. - В сб. "Вопросы проектирования базов данных", Киев, ИК, 1977.

5. Ч е р н я к Ю.И. Системный анализ в управлении экономикой. М., "Экономика", 1975.

6. Ц у ж а н д р о В.Л., Т е л и н с к о в А.М., З у б о в В.С. Принципы построения информационной базы АСУ на ЭВМ III поколения. М., ЦЭМИ. 1977.

7. S u n d g r e n B. An infological approach to data bases. Stockholm, 1973.

8. D a b r o w s k i A. O teorii informacji. Warszawa, 1974.

9. L a n g e f o r s B. Information systems. - Information Processing 74. North-Holland Publ. Co., 1974.

10. M o u l i n P., R a n d o n J. etc. Conceptual model as a data base design tool. - Modelling in Data Base Management Systems, North-Holland Publ. Co., 1976.

On Infological Approach to Data Bases

Summary

Urgent need for data base systemetizing means which should support coordinated interaction and mutual understanding between subject-matter-oriented specialists and computer-oriented specialists is pointed out.

Infological approach to data bases, basic concepts of infological theory and infological intermediary model are introduced.

Object-system - data base mapping as a set of homomorphic submappings (object-system - subject-matter model - infological model - datalogical model - data base) is described.

№ 464

TALLINNA POLÜTEHNILISE INSTITUUDI TOIMETISED

ТРУДЫ ТАЛЛИНСКОГО ПОЛИТЕХНИЧЕСКОГО ИНСТИТУТА

УДК 512.25/26+519.3:330.115

Э.А.-Ю.Юби

СИЛЬНО СОСТОЯТЕЛЬНАЯ ОЦЕНКА РЕШЕНИЯ ОБЩЕЙ ЗАДАЧИ СТОХАСТИЧЕСКОГО ПРОГРАММИРОВАНИЯ

I. Постановка задачи

Рассмотрим общую задачу стохастического программирования

$$F_0(x) \longrightarrow \min,$$
$$F_i(x) = M f_i(x,y) \leq 0, \quad i = 1, \dots, m, \tag{I}$$
$$x \in X \subset R^n, \; y \in Y \subset R^s,$$

где y - s-мерная случайная величина, закон распределения которой не зависит от x. Задачу (I) заменяем последовательностью задач (2).

$$F_0(x) \longrightarrow \min,$$
$$\hat{F}_i(x, y^i, L) = \frac{\sum_{k=1}^{L} f_i(x, y^{ik})}{L} \leq 0, \quad i = 1, \dots, m, \tag{2}$$
$$x \in X,$$

где $y^{ik} \in Y$ независимые при всех i и k реализации случайной величины y, $i = 1, \dots, m$, $k = 1, \dots, L$, $L = 1, 2, \dots$

В данной работе предлагается для приближенного решения задачи (I) заменить ее задачей (2), а решение последней брать в качестве оценки решения исходной задачи при достаточно больших L.

Найдены условия, когда с вероятностью I

$$\hat{x}(N) = \hat{x}(y^{11}, \dots, y^{mL}) \longrightarrow x^*, \quad N \longrightarrow \infty, \tag{3}$$

где x^* - решение задачи (I), а $\hat{x}(N) = \hat{x}(y^{11}, \dots, y^{mL})$ - задачи (2), $N = mL$.

В работе [I] предложен для решения задачи (I) метод, основанный на замене задачи (I) задачей (2) и сочетающий идеи методов нелинейного программирования и статистического оценивания. В данной статье приводятся подробные доказательства утверждения работы [I].

Ю.М. Ермольев [2] предложил для решения задачи (I) стохастический метод штрафов (с. 2I2). А.М. Гупал [3] решает задачу (I) методом, который сочетает идеи метода Б.Т.Поляка решения задач нелинейного программирования и Кифера-Вольфовица минимизации функции регрессии.

Предложенный в данной работе метод можно условно называть полупрямым. Как в прямых методах используются только реализации случайных параметров, по аналогии с непрямыми методами закон распределения случайных параметров задачи считается заданным, равным эмпирическому закону.

Идея решения задачи (I) путем замены функций регрессии эмпирическими функциями регрессии практически просто реализуема в том случае, когда для решения задачи (2) можно применить уже существующие алгоритмы.

Замечание I. Если решение x^* задачи (I) существует, то при некоторых реализациях y^{ik} задача (2) может и не иметь решения. В таком случае для определенности положим $\hat{x}(N)=0$. При выполнении условий последующих теорем 2 и 3 вероятность того, что множество допустимых решений задачи (2) пустое, стремится к 0 при $N \to \infty$.

Замечание 2. Задачи с вероятностными и квантильными ограничениями также могут быть преобразованы к виду (I). Если целевая функция - математическое ожидание некоторой случайной величины, то путем введения дополнительной переменной и дополнительного ограничения всегда можно добиться того, чтобы значения целевой функции вычислялись точно. Поэтому всегда можно считать, не умаляя общности, что точный аналитический вид целевой функции $F_0(x)$ известен.

2. Задача нелинейного стохастического программирования

Обозначим через Q множество допустимых решений задачи (I). Допустим, что эта задача имеет единственное решение x^* и $F_0(x)$ непрерывная функция, аналитический вид которой известен.

В работе [4] используется классическая схема аппроксимации множества допустимых решений для получения условий непрерывности оптимального решения задачи (I) при возмущении функций ограничений $F_i(x)$ $(i=1,\ldots,m)$. Непрерывное изменение точки минимума при расширении множества допустимых решений гарантируется условием A, при сужении — условием B (см. также [I]).

Условие A. Пусть $x_k \in X \setminus Q$, $x_k \to \bar{x}$, $k \to \infty$. Если для любой такой последовательности $\{x_k\}$ $\underline{\lim}\, F_i(x_k) \leqslant 0$ для всех i, $i=1,\ldots,m$ одновременно, то $F_0(\bar{x}) > F_0(x^*)$.

Пусть $U=\{x: x\in X,\ F_i(x)<0,\ i=1,\ldots,m\}$.

Условие B. Замыкание U содержит точку x^*, $x^* \in \bar{U}$.

Условие C. Функция $f_i(x,y)$ удовлетворяет условию C в области $H \subset R^n$, если $\hat{F}(x,y^i;L)$ сходится к $F_i(x)$ равномерно по $x\in H$ при $L\to\infty$, т.е. для любого $\varepsilon>0$

$$\alpha_L[f(x,y)] = P\{\text{хотя бы при одном } M \geqslant L\, |\hat{F}_i(x,y^i;M)-F_i(x)|>\varepsilon\} \to 0$$

равномерно по $x\in H$ при $L\to\infty$.

Теорема I. Пусть выполняются условия

I) множество X — компакт в R^n, а x^* — единственная точка минимума непрерывной функции $F_0(x)$ в Q,

2) условие A выполняется,

3) существует $\varepsilon_0>0$, такое, что если $y^{i1},\ldots,y^{iL}$ удовлетворяют условию $|F_i(x)-\hat{F}_i(x,y^i;L)|<\varepsilon$ для всех $x\in X$ и $i=1,\ldots,m$, то множество допустимых решений $\hat{Q}(N)$ задачи (2) компакт для любого ε, $0<\varepsilon\leqslant\varepsilon_0$,

4) все функции $f_i(x,y)$ $(i=1,\ldots,m)$ удовлетворяют условию C в X, тогда существует подпоследовательность $\hat{x}(N_k)$ решений задачи (2), что при $N_k\to\infty$ $\hat{x}(N_k)\to x^*$ и $F_0(\hat{x}(N_k))\to F_0(x^*)$ с вероятностью I.

Доказательство. Обозначим через $I=\{1,\ldots,m_1\}$ множество индексов i, $m_1\leqslant m$, что $F_i(x^*)=0$. Для $i\in I$ будем принимать во внимание только такие y^{ik}, что $\hat{F}_i(x^*,y^{ik};L_k)\leqslant 0$. Не будем накладывать никаких условий на $y^{i\cdot}$ для $i\in I$. Если условия задачи (2) дополнять ограничением $\hat{F}_i(x^*, y^{i\cdot}, L)\leq F_i(x^*)$, $i\in I$, то из последовательности решений $\hat{x}(N)$ задачи выделяется подпоследовательность $\hat{x}(N_k)$, существование которой утверждается в теореме, $\hat{x}(N_k)\longrightarrow x^*$ с вероятностью I, $N_k\longrightarrow\infty$. По лемме Бореля-Кантелли с вероятностью I происходит бесконечно много событий $\hat{F}_i(x^*, y^{i\cdot}, L)\leq F_i(x^*)$ для всех $i\in I$.

Докажем, что $P[x^*\in\hat{Q}(N_k)]\to 1$, $N_k\to\infty$. Пусть $F_i(x^*)=a_i<0$, $a=\max a_i$, $i\in I$. Тогда по усиленному закону больших чисел из условия 4 теоремы следует, что для любого ε, $0<\varepsilon<a$, при $i\in I$ $P\{|\hat{F}_i(x^*; y^{i_k\cdot} L_k)-F_i(x^*)|>\varepsilon\}\longrightarrow 0$, $L_k\to\infty$. Для $i\in I$ по условию $\hat{F}_i(x^*, y^{i_k\cdot}, L_k)\leq F_i(x^*)=0$ и этим ограничениям точка x^* удовлетворяет всегда. Следовательно, $P\{x^*\in\hat{Q}(N_k)\}\longrightarrow 1$, $N_k\longrightarrow\infty$.

Рассмотрим множество $\tilde{Q}$, $\tilde{Q}_i=\{x: x\in X$, существует $y\in Y$, что $f_i(x,y)\leq 0\}$, $\tilde{Q}=\bigcap_{i=1}^{m}\tilde{Q}_i$.

Возможны два случая.

I) $$\inf_{x\in\tilde{Q}\setminus Q} F_0(x)>F_0(x^*). \qquad (4)$$

Если

а) $f_i(x^*,y)\leq 0$ для всех $y\in Y$ и $i=1,\ldots,m$, то это означает, что при любых реализациях y точка x^* допустима и при выполнении (4) и оптимальна. Тогда $\hat{x}(N_k)\equiv x^*$ и утверждение теоремы очевидно выполняется.

б) существует $y\in Y$, что $f_i(x^*,y)>0$, $i\in I$. Тогда при некоторых реализациях y точка x^* недопустима.

Надо доказать, что для любого $\varepsilon>0$ $\pi_{N_k}=P\{$ хотя бы при одном $M\geq N_k\ \|\hat{x}(M)-x^*\|>\varepsilon\}\to 0$, $N_k\to\infty$. Пусть $A_{N_k}=\{$существует $i\in I$, что хотя бы при одном $M\geq N_k$ $\hat{F}_i(x^*, y^{i_k\cdot}, M)>0\}$.

$\pi_{N_k}=P\{$ хотя бы при одном $M\geq N_k\ \|\hat{x}(M)-x^*\|>\varepsilon\}\leq P\{x^*\bar{\in}\hat{Q}(M)$ хотя бы при одном $M\geq N_k\}=P(A_{N_k})\leq P\{$ существует $i\in I$, что хотя бы при одном $M\geq N_k\ |\hat{F}_i(x^*, y^{i_k}; M)-F_i(x^*)|>\varepsilon_0, 0<\varepsilon_0<-a\}\longrightarrow 0$, так как для всех i $\hat{F}_i(x^*, y^{i_k\cdot}, L_k)\longrightarrow F_i(x^*)$ с вероятностью I, $N_k\longrightarrow\infty$ для всех $\varepsilon_0, \varepsilon_0>0$, в том числе и для $0<\varepsilon_0<-a$.

Следовательно, $\hat{x}(N_к) \to x^*$ с вероятностью I при $N_к \to \infty$.

2) $\inf\limits_{x \in \tilde{Q} \setminus Q} F_0(x) \leq F_0(x^*)$. (5)

Докажем, что $\pi_{N_к} = P\{$ хотя бы при одном $M \geq N_к \, \| \hat{x}(M) - x^* \| > \varepsilon \} \to 0$, $N_к \to \infty$. Очевидно, $\pi_{N_к} \leq P(A_{N_к} + B_{N_к})$, где

$B_{N_к} = \{$ существует хотя бы один $M \geq N_к$ и $x_м$, что $x_м \in U_\varepsilon$ и $x_м \in \hat{Q}(M)$, $U_\varepsilon = \{x : x \in X \setminus Q,\ \|x - x^*\| > \varepsilon,\ F_0(x) \leq F_0(x^*)$,

$U_\varepsilon(i) = \{x : x \in U_\varepsilon,\ F_{m+1}(x) = F_i(x)\}$.

Докажем, что $P(A_{N_к}) \to 0$, $P(B_{N_к}) \to 0$, тогда и $\pi_{N_к} \to 0$, $N \to \infty$. В пункте б) первой части было доказано, что $P(A_{N_к}) \to 0, N_к \to \infty$. По доказанному в работе [4] $\inf\limits_{x \in U_\varepsilon} F_{m+1}(x) = a_\varepsilon > 0$, где

$F_{m+1}(x) = \max\limits_{1 \leq i \leq m} F_i(x)$. Очевидно, что $B_{N_к} \subset \sum\limits_{i=1}^{m} C_{N_к}(i)$. $C_{N_к}(i) = \{$ существует хотя бы один $M \geq N_к$ и $x_m \in U_\varepsilon(i)$, что $|F_i(x_м) - \hat{F}_i(x_м, y^{i}, \frac{M}{m})| > \varepsilon_0$, $0 < \varepsilon_0 < a_\varepsilon \}$. Для доказательства теоремы достаточно показать, что $P(C_{N_к}(i)) \to 0$, $N_к \to \infty$, $i = 1, \ldots, m$. Пусть, например, $U_\varepsilon(1) \neq \emptyset$. $P(C_{N_к}(1)) = P\{$ существует хотя бы один $M \geq N_к$ и $x_м \in U_\varepsilon(1)$, что $|F_1(x_м) - \hat{F}_1(x_м, y^1, M/m)| > \varepsilon_0\} \to 0, N_к \to \infty$, $0 < \varepsilon_0 < a_\varepsilon$, так как в области $U_\varepsilon(1)$ выполняется условие C для функции $f_1(x, y)$.

Теорема доказана.

Замечание 3. Условие 3 теоремы выполняется, например, когда $F_i(x)$ $(i = 1, \ldots, m)$ полунепрерывны снизу, или выполняются условия леммы I работы [I]. Если выполняется условие 3 теоремы и условие C для $f_i(x, y)$ $(i = 1, \ldots, m)$ в X, то очевидно, вероятность того, что $\hat{Q}(N_к)$ компакт стремится к I при $N_к \to \infty$.

Замечание 4. Пусть H ограниченная область, $H \subset R^n$. Докажем, что если $M|f(x, y)| \leq c$ для всех $x \in H$, то в области H выполняется условие C, $\alpha_N[f(x, y)] = P\{$ хотя бы при одном $M \geq N \mid F(x) - \hat{F}(x, y, M)| > \varepsilon\} \to 0$ равномерно по $x \in H$ при $N \to \infty$. Действительно, так как $\alpha_N[f(x, y)] \geq \alpha_{N+1}[f(x, y)]$ при любом $N = 1, 2, \ldots$, то достаточно доказать, что не существует последовательности $x_к \in H$, $к = 1, 2, \ldots$, что при любом фиксированном $N \alpha_N[f(x_k, y)] \to 1$, $k \to \infty$. Допустим противное, что такая последовательность существует. По ограниченности множества H существует сходящаяся подпоследовательность $\{x_j\}$, $x_j \to \bar{x}$, $x_j \in H$, $\alpha_N[f(x_j,y)] \to 1$, $j \to \infty$ при любом фиксированном N. Положим $\eta(y) = \underline{\lim}|f(x_j,y)|$, $j \to \infty$. По лемме Фату $M\eta(y) = M\underline{\lim}|f(x_j,y)| \leq \underline{\lim} M|f(x_j,y)| \leq c$ и функция $\alpha_N[\eta(y)]$ определена для $\eta(y)$, $\alpha_N[\eta(y)] \equiv 1$ при любом N. Но из существования математического ожидания $M\eta(y)$, $M\eta(y) \leq c$ следует, что $N^{-1}\sum_{k=1}^{N}\eta(y^k) \to M\eta(y)$ с вероятностью I и $\alpha_N[\eta(y)] \to 0$, $N \to \infty$. Мы получим противоречие с тождеством $\alpha_N[\eta(y)] \equiv 1$ для всех N.

Замечание 5. Условие $M|f(x,y)| \leq c$ для всех $x \in H$ выполняется, если для всех $x \in H$ выполняется

а) $Df(x,y) \leq c_1$,

б) $|f(x,y)| \leq c_2$ для всех $y \in Y$,

в) $Mf(x,y) - f(x,y)$ не зависит от x и $Mf(x,y) \leq c_3$.

Необходимое достаточное условие независимости $Mf(x,y) - f(x,y)$ от x - сепарабельность $f(x,y)$, $f(x,y) = f_1(x) + f_2(y)$.

Замечание 6. Условие $Df_i(x,y) < \infty$, $x \in X \setminus Q$ может и не выполняться, а тем не менее сходимость $\hat{x}(N) \to x^*$ с вероятностью I при $N \to \infty$ может иметь место, как следует из примера I.

<u>Пример I.</u>

$$F_0(x) = -x \to \min,$$
$$F_1(x) = M(x+y) = x \leq 0,$$
$$X = [-10, 10],$$

где $My = 0$, $Dy = \infty$,

$$N = L,\ \hat{y}^1_N = (y^{11} + \cdots + y^{1N})/N, \qquad \hat{x}(N) = \begin{cases} 10, & \hat{y}^1_N \leq -10 \\ -\hat{y}^1_N, & -10 \leq \hat{y}^1_N \leq 10, \\ -10 & 10 \leq \hat{y}^1_N. \end{cases}$$

Очевидно, $\hat{x}(N) \to x^* = 0$ с вероятностью I при $N \to \infty$, так как существует математическое ожидание $M(-y) = 0$.

Пример 2.
$$
\begin{gathered}
-9x_1-8x_2 \to \min,\\
-x_1+2x_2 \leqslant Mb_1,\\
4x_1+3x_2 \leqslant Mb_1,\\
x_1-4x_2 \leqslant Mb_3,\\
x_1 \geqslant 0,\quad x_2 \geqslant 0,\\
b_1 \in N(10,1),\ b_2 \in N(60,5),\ b_3 \in N(8,2).
\end{gathered}
$$

Легко проверить, что $x^*=\left(\frac{90}{11}, \frac{100}{11}\right)$,

$$
\hat{x}(N)=\begin{pmatrix}\hat{x}_1(N)\\ \hat{x}_2(N)\end{pmatrix}=\begin{pmatrix}-\frac{3}{11} & \frac{2}{11}\\ \frac{4}{11} & \frac{1}{11}\end{pmatrix}\begin{pmatrix}\hat{b}_1(N)\\ \hat{b}_2(N)\end{pmatrix}, \text{ если}
$$

$$
\begin{aligned}
-19\hat{b}_1(N)-2\hat{b}_2(N)-11\hat{b}_3(N) &\leqslant 0,\\
-3\hat{b}_1(N)+2\hat{b}_2(N) &\geqslant 0,\\
4\hat{b}_1(N)+\hat{b}_2(N) &\geqslant 0.
\end{aligned}
$$

Если в добавок к условиям теоремы I выполняется условие В, то вся последовательность $\hat{x}(N)$ сходится к x^* с вероятностью I при $N\to\infty$. В таком случае можно для оценки всех средних значений $Mf_i(x,y)$ использовать одни и те же независимые реализации $y^1,\ldots,y^N,\ldots$ Это доказывается в следующей теореме.

Пусть $y^1,\ldots,y^N,\ldots$ независимые реализации случайной величины y, $y^k\in Y$. Рассмотрим задачу

$$
\begin{gathered}
F_0(x)\to\min,\\
\hat{F}_i(x,y,N)=\frac{\sum_{k=1}^{N} f_i(x,y^k)}{N}\leqslant 0,\ i=1,\ldots,m,\\
x\in X.
\end{gathered} \tag{6}
$$

Последующая теорема аналогична теореме 2 из [4].

Теорема 2. Пусть выполняются условия

I) множество X компакт в R^n, а x^* единственная точка минимума непрерывной функции $F_0(x)$ в Q,

2) условие А выполнено,

3) существует $\varepsilon_0>0$ такое, что если $y^1,\ldots,y^N$ удовлетворяют условиям $|F_i(x)-\hat{F}_i(x,y,N)|<\varepsilon$ для всех $x\in X$ и $i=1,\ldots,m$, то множество допустимых решений $\hat{Q}(N)$ задачи (6) компакт для любого ε, $0<\varepsilon\leqslant\varepsilon_0$,

4) все функции $f_i(x,y)(i=1,\ldots,m)$ удовлетворяют условию

C в X.

5) выполняется условие B,

тогда $\hat{x}(N) \to x^*$ и $F_0(\hat{x}(N)) \to F_0(x^*)$ с вероятностью I, $N \to \infty$.

Доказательство. В случае I) при выполнении а) доказательство проводится аналогично доказательству предыдущей теоремы. Рассмотрим, когда

б) существует $y \in Y$, что $f_i(x^*, y) > 0$.

Докажем, что для любого $\varepsilon > 0$

$\pi_N = P\{$ хотя бы при одном $M \geq N \ \|\hat{x}(M) - x^*\| > \varepsilon\} \to 0$, $N \to \infty$.

Рассмотрим последовательность $\{x_k\}$, существование которой следует из выполнения условия B. Пусть k_1 такой номер, что $\|x_{k_1} - x^*\| < \varepsilon$, $F_i(x_{k_1}) = b_i < 0$ для всех $i = 1, \ldots, m$, $b = \max b_i$, $i = 1, \ldots, m$, $\hat{F}_i(x_{k_1}, y, N) \to F_i(x_{k_1})$ с вероятностью I при $N \to \infty$, поэтому и для всех $i = 1, \ldots, m$ $P\{$существует хоть один $M \geq N$, что $|\hat{F}_i(x_{k_1}, y, M) - F_i(x_{k_1})| > \varepsilon_1\} \to 0$, $N \to \infty$ для всех $\varepsilon_1 > 0$. в том числе и для ε_1, $0 < \varepsilon_1 < -b$. Поэтому $P(A_N) \to 0$, $N \to \infty$, где событие $A_N = \{$ существует i, $1 \leq i \leq m$, что хотя бы при одном $M \geq N$ $\hat{F}_i(x_{k_1}, y, M) > 0\}$. Очевидно, что $\pi_N \leq P(A_N)$ и $\pi_N \to 0$, $N \to \infty$.

2) $$\inf_{x \in \tilde{Q} \setminus Q} F_0(x) \leq F_0(x^*).$$

Очевидно, что $\pi_N \leq P(A_N + B_N)$, где A_N определяется как и в первой части доказательства, а B_N – в доказательстве предыдущей теоремы. Тогда $P(A_N) \to 0$, $P(B_N) \to 0$ и следовательно, $\pi_N \to 0$, $N \to \infty$. Теорема доказана.

3. Задача линейного стохастического программирования

Рассмотрим задачу линейного стохастического программирования

$$
\begin{aligned}
&F_0(x) = cx \to \min, \\
&F_i(x) = Ma_{i\cdot}x - Mb_i \leq 0, \quad i = 1, \ldots, m, \qquad (7)\\
&x \in R^n_+ = \{x : x \in R^n,\ x_k \geq 0,\ k = 1, \ldots, n\},
\end{aligned}
$$

где $c = (c_1, \ldots, c_n)$ – детерминированный, b_i, $a_{i\cdot} = (a_{i1}, \ldots, a_{in})$ $(i = 1, \ldots, m)$ – случайные. Допустим также, что ни один из векторов $a_{i.}$ не равняется нулю. Для оценки средних значений Ma_{ij}, Mb_i используем по аналогии с предыдущим разделом величины

$$\hat{a}_{ij} = \frac{\sum_{k=1}^{N} a_{ij}^{k}}{N}, \quad \hat{b}_i = \frac{\sum_{k=1}^{N} b_i^{k}}{N}, \quad i = 1,\ldots,m, \; j = 1,\ldots,n,$$

где a_{ij}^{k} и b_i^{k} независимые реализации случайных величин a_{ij}, b_i. Решение x^* задачи (7) оцениваем с помощью решения $\hat{x}(N)$ задачи

$$\begin{gathered} F_0(x) = cx \to \min, \\ \hat{F}_i(x, \hat{a}_{i.}, N) = \hat{a}_{i.}x - \hat{b}_i \leq 0, \quad i = 1,\ldots,m, \\ x \in R_+^n. \end{gathered} \tag{8}$$

Теорема 3. Пусть выполнены условия

I) x^* единственная точка минимума cx в Q,

$Q = \{x: Ma_{i.}x - Mb_i \leq 0, \; i = 1,\ldots,m, \; x \in R_+^n\}$,

причем предполагается, что существуют все математические ожидания $M|a_{ij}| < \infty$, $M|b_i| < \infty$, $i = 1,\ldots,m$, $j = 1,\ldots,n$,

2) множество Q имеет внутреннюю точку, тогда $\hat{x}(N) \to x^*$ с вероятностью I при $N \to \infty$.

Доказательство.

Рассмотрим компакт K, $K = \{x: 0 \leq x_j \leq T, \; j = 1,\ldots,n\}$, причем $x_j^* < T$, $j = 1,\ldots,n$. Докажем, что для любого $\varepsilon > 0$ все функции $a_{i.}x - b_i$ $(i = 1,\ldots,m)$ удовлетворяют условию C в K, $\alpha_N[a_{i.}x - b_i] = P\{$хотя бы для одного $M \geq N \mid (\hat{a}_{i.}(M) - Ma_{i.})x - \hat{b}_i(M) + Mb_i| > \varepsilon\} \leq P\{$ хотя бы для одного $M \geq N|(\hat{a}_{i.}(M) - Ma_{i.})\tilde{x} - \hat{b}_i(M) + Mb_i| > \varepsilon\} \to 0$, $N \to \infty$, где $\tilde{x}$ - некоторая граничная точка K. Следовательно, условие 4 теоремы 2 для $X = K$ выполняется. Из условия 2 следует, что выполняется условие B для $X = K$. Для компакта K выполнены все условия теоремы 2 и оптимальное решение задачи (8) с дополнительным условием $x \in K$ сходится с вероятностью I к x^* при $N \to \infty$. Очевидно, что тогда решение $\hat{x}(N)$ задачи (8) тоже сходится с вероятностью I к x^* при $N \to \infty$.

Теорема доказана.

Замéчание 7. Если множество Q не имеет внутренней точки, то можно выделить подпоследовательность $\hat{x}(N_k) \to x^*$ с вероятностью I при $N_k \to \infty$.

Замечание 8. В случае, когда только b_i случайные и в добавок к условиям теоремы 3 дисперсии $Db_i < \infty$ для всех i, $1 \leq i \leq m$, то $\hat{x}(N)$ является наилучшей линейной оценкой для x^*. Это следует из формулы Крамера.

Замечание 9. Рассмотрим более подробно задачу (7), в которой только правые части b_i случайные. Последующие рассуждения в этом замечании носят неформальный характер,формулировка понятий и точные утверждения не приводятся.

Допустим, что выполняются предположения теоремы 3 и пусть еще Q компакт.

В работе [5] введено понятие ε-устойчивости задач стохастического программирования. Ограничения рассматриваются в виде $a_{i\cdot}x \leq b_i(\xi)$, где $b_i(\xi)$ реализации случайных правых частей. Задача называется ε-устойчивой, если с вероятностью не меньшей $1-\varepsilon$ оптимальному решению соответствует базис, соответствующий x^*, иными словами, базис, соответствующий оптимальному решению по средним Mb (б.с.о. р.ср.). Для решения задачи (7) (задачи по средним Mb)нужно, во-первых, определить б.с.о.р.ср, и во-вторых, оценить Mb через $\hat{b}(N)$. Определение б.с.о.р.ср. заменим проверкой гипотезы о том, что среднее значение Mb m-мерной случайной величины b принадлежит области устойчивости D(т.е. области, в которой могут меняться правые части, чтобы базисом остался б.с.о. р.ср., см. [5]).

Пусть для оптимального решения по средним x^* r $(r \leq m)$ первых ограничений задачи (7) выполняются как равенства и $x^*_k \neq 0$, $k=1,\dots,r$, $x^*_k = 0$, $k=r+1,\dots,m$.Обозначим через B соответствующую базисную матрицу. Тогда $x^*_k = B^{-1}_k \tilde{M}b$, где B^{-1}_k – k-я строка обратной матрицы,а $\tilde{M}b$ вектор из r первых компонент Mb, $k=1,\dots,r$. Выпишем в пространстве R^m правых частей $b_i (i=1,\dots,m)$ уравнения m гиперплоскостей, которые ограничивают область устойчивости D. В пространстве R^m получим уравнения $m-r$ гиперплоскостей,если при изменении правых частей какое-то из $m-r$ небазисных ограничений превращается в равенство. В уравнение небазисного ограничения $\sum_{k=1}^{r} a_{ik} x_k + \sum_{k=r+1}^{m} a_{ik} \cdot 0 = Mb_i$ подставим

$x_k = B^{-1}_k \tilde{b}$, $i=r+1,\dots,m$, $k=1,\dots,r$. Получим уравнения $m-r$ гиперплоскостей $\sum_{k=1}^{r} a_{ik}(B_k^{-1}\tilde{b}) = Mb_i$, $i = r+1,\dots,m$. Уравнения остальных r гиперплоскостей получим, когда правые части выходят из области устойчивости по той причине, что $x_k = B_k^{-1}\tilde{b}$ становятся отрицательными, $B_k^{-1}\tilde{b} = 0$, $k=1,\dots,r$. Итак, пространство R^m правых частей b_i $(i=1,\dots,m)$ разделяется m гиперплоскостями

$$\sum_{k=1}^{r} a_{ik}(B_k^{-1}\tilde{b}) = b_i, \; i = r+1,\dots,m, \qquad (9)$$
$$B_k^{-1}\tilde{b} = 0, \; k = 1,\dots,r,$$

на не более чем 2^m областей, в одной из которых находится точка Mb. Если мы нормируем уравнения (9) и подставим в них координаты $\hat{b}(N)$, получим расстояния от точки $\hat{b}(N)$ до граничных гиперплоскостей области устойчивости (будем считать расстояние от гиперплоскости в одном полупространстве положительным, в другом – отрицательным). При решении задачи (8) по оценке $\hat{b}(N)$ базис совпадает с б.с.о.р.ср., если расстояние от точки $\hat{b}(N)$ до всех граничных гиперплоскостей области устойчивости D того же знака, что и от точки Mb (см. пример 2.).

Предлагается двухэтапная процедура для решения задачи (7). На первом этапе решается задача (8). Выписываются границы области устойчивости для случая, когда б.с.о.р.ср. совпадает с базисом, который соответствует $\hat{x}(N)$ -оптимальному решению задачи (8). Вычисляется отношение S/T, где S – сумма выборочных дисперсий b_i, $i=1, \dots m$, T_i -расстояние от точки $(\hat{b}_1, \dots, \hat{b}_m)$ до i-й гиперплоскости (9), $T = \min T_i$, $i=1,\dots,m$. При близком к нулю S/T б.с.о.р.ср. считается базис, соответствующий $\hat{x}(N)$. В противном случае (например, когда $S/T > (0,5)$ нужно еще наблюдать реализации правых частей, увеличить N. Когда точка средних Mb находится близко к границе области устойчивости, проверка гипотезы затруднительна.

Литература

I. Ю б и Э.А.-Ю. Статистическое исследование и метод решения задач стохастического программирования." Изв. АН ЭССР. Физика. Математика", 1977, т. 26, № 4, с. 369-376.

2. Ермольев Ю.М. Методы стохастического программирования. М., "Наука", 1976, 240 с. с ил.

3. Гупал А.М. Об одной задаче стохастического программирования с ограничениями вероятностной природы. "Кибернетика", 1974, № 6, с. 94-100.

4. Юби Э.А.-Ю. Вертикальная устойчивость задач нелинейного программирования. Рукопись депонирована во ВИНИТИ. 8.12.1977. № 4409-77 Деп. 16 с.

5. Арбузова Н.И. О стохастической устойчивости двойственных задач линейного программирования. "Экономика и математические методы", 1966, т.2, № 4, с. 558-562.

E. Übi

The Consistent Estimate of the Solution of the General Stochastic Programming Problem

Summary

The stochastic programming problem where the mean constraints are replaced by their estimates is investigated. It is proved that under certain conditions the estimate of the solution is consistent.

№ 464
TALLINNA POLÜTEHNILISE INSTITUUDI TOIMETISED

ТРУДЫ ТАЛЛИНСКОГО ПОЛИТЕХНИЧЕСКОГО ИНСТИТУТА

УДК 518.9:65.012

И.Э.Муллат

ПРИЛОЖЕНИЕ МОНОТОННЫХ СИСТЕМ К ИЗУЧЕНИЮ СТРУКТУРЫ МАРКОВСКИХ ЦЕПЕЙ

Настоящая работа посвящена приложению развитой в [I] теории монотонных систем к марковским цепям. С одной стороны интерес к марковским цепям вызван тем, что на них удобно интерпретировать специальный класс поглощающих цепей как монотонную систему, а с другой - существует возможность иллюстрации основных свойств подобной монотонной системы на примере коммутируемой телефонной сети.

С целью объяснения на содержательном уровне развитого в данной работе аппарата выделения экстремальных подсистем в марковских цепях сперва приведем в несколько видоизмененной форме пример монотонной системы - телефонной сети коммутаций. Затем показывается, как марковская цепь может быть ассоциирована с этой монотонной системой и какие принципиальные операции можно осуществлять на марковской цепи, с тем, чтобы использовать теоретический аппарат монотонных систем, изложенный в [I].

В работе [I] рассматривался пример коммутируемой телефонной сети в виде множества W линий связей между пунктами связи. Предположим, теперь, что каждая линия связи состоит из основного и резервного канала. В случае отсутствия прямой линии между какими-либо пунктами в [I] предполагалось, что контакты коммутируются транзитным путем. В дополнение к этому не исключена возможность коммутирования контактов по транзитным путям даже в том случае, когда между пунктами имеется прямая линия связи.

В работе [I] каждая пара пунктов характеризовалась средним числом "отказов" в установлении связи. Число отказов обычно характеризует линии связи в коммутируемых телефонных сетях. В излагаемой ниже модели и для преследуемых здесь целей более удобной является величина, противоположная числу отказов и характеризующая нагрузку на линию связи.

Допустим, что каждая линия связи (основной и резервный канал) характеризуется пропускной способностью c_{ij} или, другими словами, максимально допустимой нагрузкой. Величина c_{ij} учитывает пропускную способность основного и резервного каналов. Функционирование пункта связи s описывается максимально допустимой нагрузкой $c_s = \sum_{j=1}^{n} c_{sj}$.

Основной канал линии связи между пунктами s и j, так же как и резервный, описывается, следовательно, долей p_{sj} от приходящейся на него максимальной нагрузки c_s. В реально коммутируемой телефонной сети эта доля нагрузки должна быть меньше, так как максимально допустимая доля нагрузки c_{sj}/c_s вряд ли осуществима. Доля нагрузки p_{sj} на канал связи может интерпретироваться как вероятность коммутации контактов между s-м и j-м пунктами. Если предположить, что основной и резервный каналы тождественны, то величина

$$2\sum_{j=1}^{n} p_{sj} < 1 \tag{I}$$

для любого s.

Пусть коммутируемая телефонная сеть с указанными выше транзитными путями коммутации функционирует в течение длительного времени путем введения в действие основных каналов связи. Каждый основной канал (точнее пункты связи i и j) описывается средним числом $\bar{p}_{ij}$ коммутируемых контактов с учетом транзита. Понятно, что числа $\bar{p}_{ij}$ несколько больше чисел p_{ij}.

Предположим, что в каком-либо канале происходит обрыв. Происшедшее изменение в коммутируемой сети отражается в числах $\bar{p}_{ij}$ в виде их уменьшения. Допустим, что в какой-либо линии связи требуется увеличить реальную нагрузку за счет ввода в действие резервного канала. Понятно, что в этом случае произойдет увеличение всех чисел $\bar{p}_{ij}$. Так организованная коммутируемая телефонная сеть является монотонной системой.

Возникает задача: какое изменение вносит обрыв или ввод резервного канала в числа $\bar{p}_{ij}$. Для решения этой задачи необходима ее постановка на языке марковских цепей.

Рассматриваемое множество каналов связей W описывается квадратной матрицей $\| p_{ij} \|_n^n$. Если канал связи отсутствует, то $p_{ij} = 0$. Из теории марковских цепей [2] известно, что такого рода матрицы ассоциируются со множеством невозвратных состояний некоторой поглощающей цепи Маркова. На языке подобных цепей число $\bar{p}_{ij}$ интерпретируется как среднее число попаданий из пункта i в пункт j по марковской цепи, а обрыв или ввод резервного канала отражается в специальных формулах пересчета чисел средних попаданий $\bar{p}_{ij}$. На языке монотонных систем действие типа $\ominus$ - это обрыв основного канала, а типа $\oplus$ - это ввод резервного канала.

Исходя из сказанного, на специальном классе поглощающих марковских цепей можно поставить задачу выделения экстремальных подсистем - ядер, а с помощью разработанной в [I] конструктивной процедуры выделения ядер ПВЯ - осуществить поиск ядер.

В данной работе раздел I посвящен постановке задачи выделения ядер на марковских цепях. Во втором разделе показывается, что результат действий $\oplus$ и $\ominus$ на элементы переходной матрицы марковской цепи приводит к соотношениям Шермана-Моррисона [3] (см. приложение) для пересчета чисел средних попаданий.

I. Постановка задачи выделения ядер на марковских цепях

Рассматриваются однородные марковские цепи с конечным числом состояний и дискретным временем. Множество состояний обозначим через V. Задание однородной марковской цепи эквивалентно тому, что переход из состояния i в состояние j в некоторый момент времени $t+1$ не зависит от того, из какого состояния s $(s=1,2,\ldots,n)$ рассматриваемая цепь перешла в i в предшествующий момент времени t. Условную вероятность такого перехода из i в j за k единиц времени обозначим через $p(i,j,k)$ $(p(i,j,1)=p_{ij})$.

Ниже рассматриваются марковские цепи специального вида, для любых двух состояний i и j из некоторого подмножества V выполняется

$$\lim_{k\to\infty} p(i,j,k)=0.$$

Из теории марковских цепей известно, что равенство нулю указанного предела справедливо в случае, если состояние j невозвратно, и вследствие чего необходимо наличие у марковской цепи особых возвратных состояний. Не умаляя общности изложения, далее рассматриваются цепи с одним единственным возвратным состоянием, которое должно быть одновременно и поглощающим состоянием.

Поглощающие цепи, используемые далее, следующие:

I. Существует единственное поглощающее состояние $\Theta\in V$;

II. Все остальные состояния цепи невозвратны, и вероятности перехода между ними за один шаг задаются квадратной матрицей $\|p_{ij}\|_n^n$. Вероятность перехода в поглощающее состояние Θ из невозвратного состояния i за один шаг в соответствии с пп. I и II равна

$$p_{i\Theta}=1-\sum_{s=1}^{n} p_{is}.$$

Монотонная система требует определения положительных и отрицательных $(\oplus,\ominus)$ действий на элементы системы. Для этой цели мы воспользуемся понятием среднего числа попаданий $\bar{p}_{ij}$ из состояния i в j [2]. Известно, что величина $\bar{p}_{ij}$ выражается рядом

$$\bar{p}_{ij}=\sum_{k=1}^{\infty} p(i,j,k). \qquad (2)$$

Достаточное условие сходимости рядов (2) заключается в том, чтобы сумма элементов в любой строке матрицы $\|p_{ij}\|_n^n$ была меньше единицы. В дальнейшем везде рассматриваются цепи пп. I и II, удовлетворяющие этому условию.

Пусть W – множество ненулевых элементов матрицы $\|p_{ij}\|$. На множестве переходов W описанной выше марковской цепи определяются следующие действия.

Определение. Действием типа $\ominus$ на элемент системы W (ненулевой элемент матрицы $\|p_{ij}\|$) называется уменьшение соответствующей вероятности перехода за один шаг на некоторое значение Δp.

Точно также определяется действие типа $\oplus$. В этом случае вероятность перехода за один шаг, соответствующая элементу p_{ij}, увеличивается на некоторую величину Δp. В случае увеличения какого-либо ненулевого элемента матрицы $\|p_{ij}\|$ все средние числа попаданий $\bar{p}_{ij}$, исходя из простых вероятностных соображений, должны одновременно увеличиваться, а в случае уменьшения, наоборот, все числа также одновременно уменьшаются. Таким образом, введенные действия на систему W полностью удовлетворяют требованиям монотонности [I], а система W является монотонной системой.

Необходимо подчеркнуть, что в данном выше определении $\oplus$- и $\ominus$-действий на элементы матрицы $\|p_{ij}\|$ не заданы величины Δp изменения вероятностей переходов из множества W. Существуют достаточно богатые возможности их определения. Например, увеличение (уменьшение) каждой вероятности на определенную константу или то же изменение, но в зависимости от значения самой вероятности и т.д. При конкретном определении $\oplus$- и $\ominus$-действий на поглощающей марковской цепи желательно использовать содержательные представления. Ниже, на примере коммутируемых сетей описывается одно такое представление.

Пусть W – множество всех возможных переходов за один шаг среди невозвратных состояний поглощающей цепи. Переходы множества W отождествляются с ненулевыми элементами матрицы $\|p_{ij}\|$. Пусть T – некоторое подмножество множества W указанных ненулевых элементов. Обозначим через $p(T,i,j,k)$ вероятность того, что цепь переходит из состояния i в состояние j за k единиц времени при условии, что за этот период времени переходы длиной в один шаг из множества T подверглись действиям. Условие соответствует тому, что переходы по множеству W/T осуществляются в соответствии со "старыми" вероятностями, а по T – с "новыми". Не исключается возможность, что действия $\oplus, \ominus$ вообще не осуществляются (множество $T = \phi$). В этом случае при обозначении соответствующей вероятности символ T опускается.*

Средние числа попаданий из i в j при условии, что некоторые переходы из множества T подвергались действиям, выражаются рядом

$$\bar{p}(T,i,j) = \sum_{m=1}^{\infty} p(T,i,j,m). \tag{3}$$

Обратимся теперь к совокупности весовых наборов, задаваемых на монотонной системе W. Определим весовой набор $\Pi^{+}H$ на подмножестве $H \subseteq W$ как набор чисел $\{\bar{p}(\bar{H},i,j) \mid (i,j) \in H\}$ в случае, если на $\bar{H}$ оказывались положительные действия, и тот же набор $\Pi^{-}H$ в случае, если на $\bar{H}$ оказывались отрицательные действия.

В работе [I] показано, что в монотонной системе обязательно существуют экстремальные подсистемы двух видов – $\oplus$- и $\ominus$-ядра. Определенные выше числа средних попаданий $\bar{p}(\bar{H},i,j)$ позволяют сформулировать понятие $\oplus$- и $\ominus$-ядра марковской цепи.

Определение. Экстремальной подсистемой системы переходов длиною в один шаг по поглощающей цепи Маркова – $\oplus$-ядром называется система $H^{\oplus} \subseteq W$, на которой функционал

$$\max_{(i,j)\in H} \bar{p}(\bar{H},i,j) \tag{4}$$

достигает минимального значения, а $\ominus$-ядром $H^{\ominus}$-подсистема, на которой функционал

$$\min_{(i,j)\in H} \bar{p}(\bar{H},i,j) \tag{5}$$

достигает максимального значения.

Обратимся теперь к иллюстрации введенных понятий $\oplus$- и $\ominus$-ядер марковских цепей на примере коммутируемой телефонной сети, описанной в начале данной части.

Числа вероятности коммутации контактов p_{ij} (без учета транзита) между пунктами i и j $(i,j = 1,2,\ldots,n)$ позволяют построить по коммутируемой телефонной сети поглощающую цепь,

---

* Предполагается, что действия не нарушают сходимости рядов, см. условие (I).

удовлетворяющую пп. I и II. Единственное требуемое условие заключается в выполнении неравенства (I), которое, как уже указывалось, является естественным в рассматриваемой модели. В этом случае числа p_{ij} интерпретируются как вероятности перехода за один шаг, а $\bar{p}_{ij}$ – как средние числа попаданий из i в j.

Поиск $\oplus$-и $\ominus$-ядер конкретной марковской цепи, построенной исходя из коммутируемой телефонной сети, требует содержательного определения $\oplus$- и $\ominus$-действий. Вначале упоминалось, что $\ominus$-действие - это обрыв основного канала связи, а $\oplus$-действие - ввод резервного канала. На марковской цепи обрыв выражается в занулении соответствующей вероятности перехода за один шаг, а ввод резервного канала увеличивает эту вероятность в два раза. Условие (I) гарантирует, что при любых таких действиях сходимость рядов (2) и (3) не нарушается.

Исходя из вышеприведенной интерпретации марковской цепи, в виде коммутируемой телефонной сети, можно предложить следующую содержательную интерпретацию $\oplus$- и $\ominus$-ядер марковской цепи.

В экстремальной подсистеме $H^{\oplus}$ все линии связи оставлены без изменения, а не принадлежащие $H^{\oplus}$ - снабжены резервным каналом. Экстремальное значение функционала (4) показывает, что среднее число коммутации контактов по линиям ядра $H^{\oplus}$, включая транзитные коммутации, достаточно мало. Это значит, что линии множества $H^{\oplus}$-ядра нечувствительны по отношению к организации транзитной связи. Множество линий $H^{\ominus}$-ядра обладает противоположным свойством. Основные каналы $\ominus$-ядра $H^{\ominus}$ наиболее "благоприятны" в смысле организации "хорошей" транзитной связи. По линиям $H^{\oplus}$ транзитные возможности наиболее ослаблены.

II. Весовые функции монотонной системы на марковских цепях

В первом разделе на элементы матрицы переходов за один шаг, соответствующей невозвратным состояниям, определялись $\oplus$- и $\ominus$-действия. В настоящем разделе разрабатывается аппарат, который позволяет учесть изменения, вносимые этими двумя типами действий в средние числа попаданий из невозвратного состояния i в состояние j. Исходя из положений [I] здесь описываются и выводятся конкретные весовые функции, предназначенные для формального описания монотонных систем. Прежде чем изложить основное содержание раздела, напомним кратко понятие весовой функции.

Допустим, что в системе W, которая в случае с марковскими цепями определяется как совокупность элементов матрицы $\|p_{ij}\|_n^n$, соответствующей переходам среди невозвратных состояний, выделено некоторое подмножество H: H – множество переходов за один шаг. В результате осуществления процесса последовательных действий типа $\ominus$ (см. раздел I) на элементы $\bar{H}$ ($\bar{H}$ - дополнение H до W) на множестве переходов H устанавливаются числа средних попаданий - весовой набор $\Pi^- H$. Аналогично на том же множестве последовательность $\oplus$-действий устанавливает весовой набор $\Pi^+ H$. Числа средних попаданий в обозначениях раздела I записываются как $\bar{p}(\bar{H},i,j)$ - это предельные значения для рядов (2) на ненулевых элементах переходной матрицы P, соответствующих элементам из H. Далее числа $\bar{p}(\bar{H},i,j)$ называются весовыми функциями.

Установим теперь общий вид весовых функций рассматриваемых марковских цепей в виде матричных рядов. Такое матричное представление в явной форме объясняет механизм определенных в I разделе действий на элементы монотонной системы - марковской цепи.

Весовую функцию марковских цепей можно найти с помощью рядов (2), где элементом ряда является вероятность перехода из i в j за k единиц времени, при условии, что на множестве переходов $\bar{H}$ осуществлялись действия.

Общий вид матриц вероятностей перехода марковских цепей, описанных в I разделе, следующий:

$$\left\| \begin{array}{cc} 1 & 0 \ldots 0 \\ p_{1\theta} & \\ \vdots & P \\ p_{n\theta} & \end{array} \right\| \qquad (6)$$

где θ - поглощающее состояние цепи;

$p_{i\theta}$ - вероятность перехода из i-го невозвратного состояния в поглощающее;

P - матрица переходных вероятностей между невозвратными состояниями за один шаг, размером $n \times n$.

Используя уравнения Колмогорова-Чепмена [2], элемент $p(T,i,j,m)$ ряда (3) можно найти в m-й степени матрицы (6) и следовательно, в матрице P^m.

Таким образом, совокупность рядов (3) можно записать в виде матричного ряда *

$$\bar{P}_T = I + P_T + P_T^2 + \ldots, \qquad (7)$$

где P_T - матрица, в которой на ненулевые элементы из множества T осуществлены действия типа ⊖ либо ⊕. Ниже T сокращенно называется множеством осуществленных действий.

Достаточные условия сходимости рядов (2) и (3) обеспечивают сходимость ряда (7). Если вспомнить, что весовая функция монотонной системы на множестве $H \subseteq W$ определяется посредством дополнения H (равного $\bar{H}$), которое и есть множество осуществленных действий, то весовая функция марковской цепи задается множеством значений элементов W матрицы $\bar{P}_{\bar{H}} = \| I - P_{\bar{H}} \|^{-1}$. Последняя матрица есть предел матричного ряда типа (7).

На языке фундаментальных матриц действия на элементы системы определяются как отдельные переходы от матрицы $\| I - P_T \|^{-1}$ к $\| I - P_{T \cup \alpha} \|^{-1}$. С вычислительной точки зрения этот переход трудоемок. Использовать матричное представление действий с целью организации поиска ⊕- и ⊖- ядер на основе описанных в [I] конструктивных процедур выделения ядер (ПВЯ) нецелесообразно. С тем, чтобы эффективно применить развитую теорию выделения экстремальных подсистем из монотонной системы на марковских цепях, требуется более экономный аппарат, который приводит к соотношениям Шермана-Моррисона [3].

Решения задачи учета изменений, возникающих в результате действий ⊕ или ⊖ на элементы матрицы переходных вероятностей за один шаг в фундаментальной матрице марковской цепи, можно добиться следующим способом. Допустим, что вместо старой вероятности перехода p_c между невозврат-

---

* Предполагается, что $p(T,i,j,0) = \delta_{ij}$, что отображено в единичной матрице I. В теории конечных марковских цепей [4] матрицы вида P_T называются фундаментальными матрицами.

ными состояниями i и j вставляется новая вероятность $p_н = p_c + \Delta p$, где действие Δp может быть любого знака. В случае, когда Δp положительное, определяется $\oplus$-действие, а когда Δp отрицательное - $\ominus$-действие. Изменение, вносимое одним из действий, можно рассматривать как два последовательных изменения: вероятность перехода p_c зануляется, и это зануление учитывается, а затем вероятность перехода восстанавливается с новым значением $p_н$, а изменение в фундаментальной матрице учитывается уже исходя из матрицы, полученной после первого изменения.

Соотношения, учитывающие изменение в фундаментальной матрице $\bar{P}_т$ как результат зануления некоторого элемента α в матрице $P_т$ и такие же соотношения для учета изменения в $\bar{P}_т$ в случае обратного $\oplus$-действия, приводятся в приложении I.

Для поиска экстремальных подсистем, следуя теории построения определяющих последовательностей элементов системы W с помощью процедур ПВЯ из [I], необходимы экономные и явные выражения, учитывающие изменение в матрице $\bar{P}_т$ при переходе к матрице $\bar{P}_{т\cup\alpha}$. Подобные выражения, позволяющие по $\bar{P}_т$ и величине Δp найти матрицу $\bar{P}_{т\cup\alpha}$, выводятся в приложении II на основе выражений П I.3 и П I.4.

С помощью рекуррентных соотношений из приложения П легко получить на любом множестве $H \subseteq W$ набор весов Π^+H или Π^-H, осуществляя последовательное применение выражения П 2.5 ко всем элементам множества $\bar{H}$. С точки зрения теоретического аппарата монотонных систем [I] выражения П 2.5 - переход от функции значимостей элементов системы π к π_α. Если Δp положительного знака, то строится весовой набор Π^+H, а если $\Delta p < 0$, то Π^-H.

Приложение I

Рассмотрим величину $\bar{p}(T,i,j)$, представленную рядом (3). Член этого ряда $\bar{p}(T,i,j,m)$ можно рассматривать как меру множества всех переходов длины m, ведущих из состояния i в j. Указанное множество путей можно считать объединением двух непересекающихся множеств: первое множество - переходы из i в j с обязательным хотя бы однажды переходом $\alpha \in W$, второе - множество путей из i в j, минующее этот переход α. Любой путь первого множества состоит из двух участков: участок переходов, обходящий α длиной в t переходов, и участок длиною в $m-t-1$, проходящий через α. Иными словами: участок в t переходов не использует переход α, а участок длиною в $t-1$ переходов этот переход α использует.

Введем следующее обозначение: $\bar{p}(T^0, i, j, k)$ - среднее число попаданий из i в j с матрицей переходов P_T, у которой ненулевой элемент α занулен. Используя введенные обозначения, получаем

П I.I
$$p(T,i,j,m) = p(T^0,i,j,m) + p_\alpha \sum_{t=0}^{m-1} p(T^0,i,\alpha_н,t)\, p(T,\alpha_к,j,m-t-1);$$

П I.2
$$p(T,i,j,m) = p(T^0,i,j,m) + p_\alpha \sum_{t=0}^{m-1} p(T,i,\alpha_н,t)\, p(T^0,\alpha_к,j,m-t-1),$$

где $\alpha_н$ - состояние, откуда осуществляется одношаговый переход α, а $\alpha_к$, где переход α заканчивается;

p_α - вероятность перехода α за один шаг или величина элемента α матрицы P_T.

Первое слагаемое в П I.I и П I.2 составляет вклад в величину $p(T,i,j,m)$ множества переходов, минующих переход α, а слагаемые, стоящие под знаком суммы, представляют вероятность того, что состояние $\alpha_н$ для соотношения

I Соотношение П I.2 выводится на основе того, что участок в t переходов использует переход α, а в $m-t-1$ переходов не использует, в противоположность выводу соотношений П I.I.

П I.I и $\alpha_к$ для соотношения П I.2 достигались соответственно с первым и последним переходом по α в моменты времени t и $t+1$.

Вычислим величины $\bar{p}(T,i,j)$, пользуясь разложением П I.I. Суммируя каждое из равенств П I.I от 1 до M, а затем, поменяв порядок суммирования в двойной сумме, получим, что

$$\sum_{m=1}^{M} p(T,i,j,m) = \sum_{m=1}^{M} p(T^0,i,j,m) +$$

$$p_\alpha \sum_{t=0}^{M-1} p(T^0,i,\alpha_н,t) \sum_{s=1}^{M-t} p(T,\alpha_к,j,s-1).$$

Разделив обе части последнего равенства на

$$\sum_{t=0}^{M-1} p(T^0,i,\alpha_н,t)$$

и рассмотрев последовательности $a_t = p(T^0,i,\alpha_н,t)$ и

$$b_{m-t} = \sum_{s=1}^{M-t} p(T,\alpha_к,j,s-1),$$

полагая при этом $M \to \infty$, получаем по теореме о средних Норлунда [2] для последовательностей a_n и b_n соотношение

П I.3 $$\bar{p}(T,i,j) = \bar{p}(T^0,i,j) + p_\alpha \bar{p}(T^0,i,\alpha_н)\,\bar{p}(T,\alpha_к,j).$$

Аналогичное соотношение получается с использованием разложения П I.2, а именно:

П I.4. $$\bar{p}(T,i,j) = \bar{p}(T^0,i,j) + p_\alpha \bar{p}(T,i,\alpha_н)\,\bar{p}(T^0,\alpha_к,j).$$

Приложение II

Введем следующие обозначения. Пусть $\bar{p}(T_c, i, j)$ – элемент матрицы $\bar{P}_T$, а $\bar{p}(T_н, i, j)$ – элемент матрицы $\bar{P}_{T\cup\alpha}$. Перепишем П I.3 и П I.4 с учетом этих обозначений и получим:

П 2.I $$\bar{p}(T_н, i, j) = \bar{p}(T^0, i, j) + p_н \bar{p}(T^0, i, \alpha_н)\bar{p}(T_н, \alpha_к, j);$$

П 2.2 $$\bar{p}(T_c, i, j) = \bar{p}(T^0, i, j) + p_c \bar{p}(T_c, i, \alpha_н)\bar{p}(T^0, \alpha_к, j).$$

Из соотношений П 2.I и П 2.2 следует, что новое значение среднего числа попаданий из i в j

П 2.3 $$\bar{p}(T_н, i, j) = \bar{p}(T_c, i, j) + p_н \bar{p}(T^0, i, \alpha_н)\bar{p}(T_н, \alpha_к, j) - p_c \bar{p}(T_c, i, \alpha_н)\bar{p}(T^0, \alpha_к, j).$$

Полагая в П. 2.I $i = \alpha_к$, получаем, что

$$\bar{p}(T_н, \alpha_к, j) = \bar{p}(T^0, \alpha_к, j)/(1 - p_н \bar{p}(T^0, \alpha_к, \alpha_н)),$$

а из П 2.2 при том же i

$$\bar{p}(T^0, \alpha_к, j) = \bar{p}(T_c, \alpha_к, j)/(1 + p_c \bar{p}(T_c, \alpha_к, \alpha_н)).$$

Подставляя последнее выражение в предыдущее и учитывая, что

$$\bar{p}(T^0, \alpha_к, \alpha_н) = \bar{p}(T_c, \alpha_к, \alpha_н)/(1 + p_c \bar{p}(T_c, \alpha_к, \alpha_н)),$$

получаем

П 2.4 $$\bar{p}(T_н, \alpha_к, j) = \bar{p}(T_c, \alpha_к, j)/(1 - \Delta p \bar{p}(T_c, \alpha_к, \alpha_н)).$$

Выражение П 2.I справедливо, если $T_н$ заменить на T_c и $p_н$ на p_c, а в выражении П 2.2 – если сделать обратное. Подставив в видоизмененное так выражение П 2.2 $j = \alpha_н$, получаем

$$\bar{p}(T^0, \alpha_к, j) = \bar{p}(T_c, \alpha_к, j)/(1 + p_c \bar{p}(T_c, \alpha_к, \alpha_н)).$$

Используя последние два равенства и выражение П 2.4 после ряда преобразований, выводим окончательное выражение для учета изменений в фундаментальной матрице $\bar{P}_T$ при переходе к $\bar{P}_{T\cup\alpha}$. В стандартных обозначениях раздела 2 вид окончательного выражения следующий:

П 2.5 $$\bar{p}(T\cup\alpha, i, j) = \bar{p}(T, i, j) + \Delta p \frac{\bar{p}(T, i, \alpha_н)\bar{p}(T, \alpha_к, j)}{1 - \Delta p \bar{p}(T, \alpha_к, \alpha_н)}.$$

Литература

1. Муллат И.Э. Экстремальные подсистемы монотонных систем. I, II, III. Автоматика и телемеханика № 5, 1976, с. 130-139, № 8, 1976, с. 169-178, № 1, 1977, с.109-119.

2. Чжун-Кай-Лай. Однородные цепи Маркова. М., "Мир", 1964.

3. Dinkelbach, W. Sensitivitätsanalysen und parametrische Programmierung. - Econometrics and Operations Research, XII, 1969.

4. Кемени Дж., Снелл Дж. Конечные цепи Маркова. М., "Наука", 1970.

J. Mullat

An Explorative Method to Study Markov Chain Structure

Summary

A Markov chain analysis method has been described. The method is based on the Markov chain transformation into a monotonic system and on a separation of kernels from the transformed chain.

№ 464
TALLINNA POLÜTEHNILISE INSTITUUDI TOIMETISED
ТРУДЫ ТАЛЛИНСКОГО ПОЛИТЕХНИЧЕСКОГО ИНСТИТУТА
УДК 681.3.016

Э.Х.-Т.Нунапуу

УРОВНИ АБСТРАКЦИЙ В БАЗАХ ДАННЫХ

1. Введение

Одним из факторов, препятствующих широкому внедрению ЭВМ, является специфика языка общения с машиной. В общении с ЭВМ естественно желание пользователя обращаться с привычными терминами и оперировать ими на семантическом уровне, абстрагируясь от машинного представления данных и машинных операций.

В настоящее время в сфере баз данных ведутся работы по обеспечению пользователя более естественными методами манипулирования данными. Для упрощения пользования базами данных предлагается новая архитектура СУБД (предложения комитета АНСИ СПАРК) [4]. В настоящей статье дается краткий обзор этих предложений. Краеугольным камнем в этих предложениях является понятие концептуальной схемы. Принципы ее построения рассматриваются в первой части настоящей статьи. Во второй части описывается многоуровневая структура СУБД и дается краткая характеристика каждого уровня.

Замечание.

Поскольку настоящая статья носит в основном реферативный характер, то следование терминологии разных авторов ведет к некоторой неоднородности терминов.

2. Отображение реального мира в базе данных

Следуя [4], определим базу данных как совокупность данных, которые представляют интересующие предприятие факты. База данных не является всеобъемлющим отражением реального мира, она характеризует реальный мир только с определенной и специфической стороны в зависимости от решаемых в автоматизированной информационной системе (АИС) предприятия задач.

Фундаментальными понятиями реального мира, которые существенны для баз данных, служат объекты, характеризуемые определенными свойствами, и отношения, которые могут быть между ними. Эти понятия невозможно определить формально так же, как в математике нет формального определения понятия множества. Понятия можно охарактеризовать посредством неформальных объяснений. Объект - это то, о чем что-то можно утверждать. Объекты могут быть конкретными (стул, мальчик) или абстрактными (цифровое значение 7), они могут быть простыми (не имеющими в своем составе других объектов) или комплексными.

2.I. Анализ объектной системы

Чтобы перейти от познаваемой реальности к описанию базы данных (называемой просто схемой или концептуальной схемой), надо провести анализ объектной системы.

Следуя статье [5], анализ объектной системы состоит из трех процедур (см. фиг. I):

- наименование;
- отбор;
- классификация.

Фиг. 1.

2.2. Наименование

Это процесс, в ходе которого каждый элемент реального мира получает имя. Тем самым определяют функциональное соответствие между множеством имен и множеством элементов. Далее каждому отношению во множестве отношений тоже дается однозначное имя.

3.3. Отбор

Как уже было сказано, не все аспекты реального мира (предприятия) должны отражаться в базе данных. В процессе отбора определяют интересующую предприятие часть реального мира.

2.4. Классификация

Это очень важное средство анализа, часто используемое в информационных системах. В процессе классификации определяются классы однородных по некоторым признакам объектов.

Описание классов объектов и отношений между этими классами, сопровождаемые дополнительными семантическими ограничениями, называют схемой или концептуальной схемой.

Более подробно и основательно дан процесс проведения анализа объектной системы в работе Б.Сундгрена [7].

3. Три уровня абстракций системы

В статье [6] авторы различают три уровня абстракций системы:

1) уровень абстракций, который отражает представления пользователя об информации и ее интерпретацию;

2) системный логический уровень абстракции (концептуальный уровень), который объединяет взгляды всех пользователей с разнообразными точками приложения в единую интегрированную логическую модель данных;

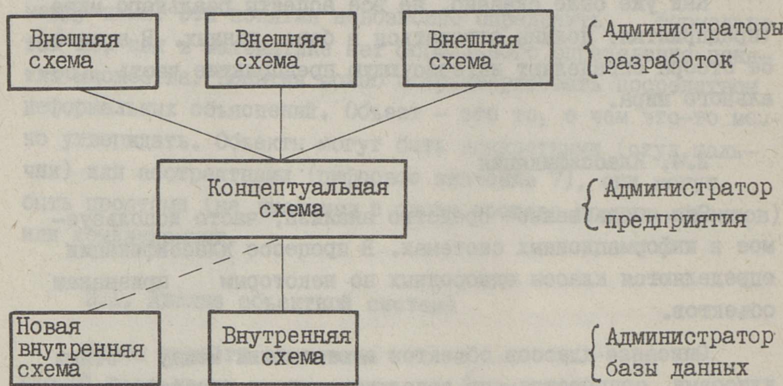
3) системный физический уровень абстракций, отражающий все детали, связанные с физической реализацией системы;

В настоящее время предлагается новая архитектура базы данных [4], в которой разделяют три уровня описания данных (см. фиг. 2):

- внешняя схема;

- концептуальная схема;
- внутренняя схема.

Они соответствуют вышеописанным уровням абстракций системы.



Фиг. 2.

3.I. Внешняя схема

В соответствии с предложениями комитета АНСИ СПАРК [4], новая архитектура баз данных дает пользователю свободу выбора подходящей модели и языка манипулирования данными. Это и есть принципиально новая черта в этих предложениях. Дополнительно к концепции независимости данных вводится еще один уровень независимости, а именно: информационная независимость .

Новая архитектура облегчает реализацию различных взглядов на одни и те же данные, образующие базу данных.

3.2. Внутренняя схема

Для того, чтобы освободить пользователей верхних (фиг. 2) уровней от несущественных для них деталей, информация о физической организации данных концентрируется во внутреннюю схему.

Внутреннюю схему составляет и изменяет по мере необходимости специалист по вычислительной технике - администратор базы данных. Обновление внутренней схемы не должно отражаться на внешних схемах. Уровень возможных изменений во внутренней схеме и определяет степень независимости данных.

4. Основные требования к концептуальной схеме

Здесь рассматриваются требования к концептуальной схеме как к связующему и центральному звену вышеописанной трехуровневой архитектуры.

Согласно статье [8], концептуальная схема должна обеспечить независимое от структуры данных описание предприятия.

Одним из требований, предъявляемых к концептуальной схеме, является ее независимость от изменений во внутренних и внешних схемах. Она должна обеспечить определение двух соответствий - между внешней и концептуальной, между концептуальной и внутренней. Значит, схема должна быть логически и физически полной. Кроме того, являясь центральным звеном системы, она должна обеспечить контроль и координацию всех доступов к определенному факту в базе данных.

Суммируя вышесказанное, можно сказать, что концептуальная схема является связующим и центральным звеном базы данных, к которому предъявляют три основных требования:

- стабильность структуры;
- контроль и координация доступа;
- физическая и логическая полнота.

Обычно [4, 8] концептуальную схему определяют как состоящую из трех секций.

Секция структуры

Здесь дается описание всех используемых любыми пользователями данных. Содержащие описание элементы должны обладать максимальной стабильностью.

Секция организаций

Здесь определяются все ограничения, которые налагаются на элементы, определенные в секции структуры.

Секция защиты

Здесь описывается, какие данные, определенные в секции структуры, являются доступными для определенного пользователя и какими функциями манипулирования он располагает.

В заключение хочется указать на трудности, возникающие при выработке концептуальной схемы. Эти трудности связаны в основном с несовместимостью функций отражения концепций предприятия с функциями, которые концептуальная схема выполняет, являющаяся связующим звеном между внутренней и внешней схемой.

Администратор предприятия желает отражать в ней только самые общие концепции предприятия, в то же время система налагает на нее требования физической и логической полноты. Другой очень трудно решаемой задачей является выработка алгоритмов определения соответствия.

По мнению автора статьи на нынешнем уровне знаний существует еще слишком много нерешенных проблем и трудностей, чтобы полностью реализовать предложения комитета АНСИ СПАРК. А изучение этих предложений имеет большую методическую ценность.

Литература

1. W o n g, H.K.T., M y l o p o u l u s, J. Two views of data semantics: a survey of data models in artificial intelligence and data base management. - INFOR, Vol. 15, No 3, Oct. 1977, pp. 344-384.

2. B i l l e r, H., N e u h o l d, E.J. The semantic interrelationship of data models. - Angewandte Informatik, 2/77, Feb. 1977, pp. 78-83.

3. S o w a, J.F. Conceptual graphs for a data base interface. - IBM J. of Res. and Devel., Vol. 20. No 4, (July 1976), pp. 336-358.

4. Interim report ANSI/X3/SPARC study group on data base management systems. ANSI, February 1975.

5. N i j s s e n, G.M. A gross architecture for the next generation DBMS. Modelling in DBMS. North-Holland P/C, 1976.

6. W a n g, C.P., W e d e k i n d, H.H. Segment synthesis in logical data base design. - IMB J. Res. and Devel., pp. 71-77. Jan. 1975.

7. S u n d g r e n, B. An infological approach to data bases. (URVAL No 7) National Central Bureau of Statistics, Stockholm. Sweden, 1973 .

8. S e n k o, M.E. Data structures and data accessing in data base systems: past, present, future. - IBM Systems Journal, Vol. 16. No 3, pp. 208-258.

9. М и х н о в с к и й С.Д. Описание внешнего пространства данных автоматизированных информационных систем. УС и М № 4, 1976, с. 22-28.

E. Õunapuu

Levels of Abstraction for Data Bases

Summary

In this paper the problems of the next generation of data base management systems are represented.

The problems connected with a conceptual schema and its design are considered there.

Содержание

ТАЛЛИНСКИЙ ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ
ТРУДЫ ТПИ № 464

АНАЛИЗ ДАННЫХ. ПОСТРОЕНИЕ ТРАНСЛЯТОРОВ
ВОПРОСЫ ПРОГРАММИРОВАНИЯ
Труды по экономике XXXУП
Редактор И. Амитан. Техн. редактор В. Ранник
Сборник утвержден коллегией Трудов ТПИ 15 ноября 1978 г.

Подписано к печати 11 мая 1979 г. Бумага 60x90/16
Печ. л. 5,75+0,25 приложение. Уч.-изд. л. 5,41
Тираж 300. МВ-04432. Ротапринт ТПИ,
Таллин, ул. Коскла, 2/9. Зак. № 505

Цена 80 коп.

Цена 80 коп.