

## МАТЕМАТИЧЕСКАЯ МОДЕЛЬ СИНТАКСИЧЕСКОГО АНАЛИЗА НЕПРОЕКТИВНЫХ И СЛАБОПРОЕКТИВНЫХ СООБЩЕНИЙ В ИНТЕЛЛЕКТУАЛЬНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЕ

Одним из путей совершенствования механизмов общения пользователей с автоматизированной информационной системой (АИС) является снятие ограничений на синтаксическую структуру входных сообщений. В частности, интеллектуальные возможности АИС могут быть существенно расширены, если обеспечить в системе синтаксический анализ непроективных и слабопроективных предложений естественного языка. В данной статье изложены результаты математического моделирования такого анализа, являющиеся развитием предложенного подхода к моделированию анализа проективных сообщений, основанного на исследовании синтагматических отношений на цепочках входного языка\*.

**Достаточные условия проективности сообщений.** Пусть  $U$  — словарь входного языка (входной словарь), элементы которого будем называть словами,  $U^*$  — множество всех цепочек, а  $U^+$  — множество всех непустых цепочек в  $U$ . Определим на цепочках входного словаря отношения парадигматического и синтагматического подчинения и синтагматической эквивалентности\*.

*Определение 1.* Отношением парадигматического подчинения назовем рефлексивное и антисимметричное отношение  $\Delta$ , если для любых цепочек  $a, b, c \in U^+$  и  $m, n \in U^*$ :

- 1)  $(mbn, c) \in \Delta$  и  $(a, b) \in \Delta$ , то  $(man, c) \in \Delta$ ;
- 2)  $(a, mbn) \in \Delta$  и  $(b, c) \in \Delta$ , то  $(a, mcn) \in \Delta$ ;
- 3)  $(a, b) \in \Delta$  и  $(c, b) \in \Delta$ , то существует цепочка  $d \in U^+$  такая, что  $(d, a) \in \Delta$  и  $(d, c) \in \Delta$ .

*Определение 2.* Отношение  $\delta$  на множестве  $U^+$  назовем отношением синтагматического подчинения, если  $(a, b) \in \delta$  тогда и только тогда, когда:

- 1) существуют цепочки  $m, n \in U^+$  такие, что  $(m, a) \in \Delta$  и  $(n, b) \in \Delta$ ;
- 2) для  $m$  и  $n$ , удовлетворяющих соотношениям п. 1, справедливо  $(m, mn) \in \Delta$ .

Цепочки вида  $ab$ , где  $(a, b) \in \delta$  или  $(b, a) \in \delta$ , будем называть синтагматическими структурами, а при  $a, b \in U$  — синтагмами. Если  $(a, b) \in \delta$ , то будем говорить, что  $a$  — определяемый, а  $b$  — определяющий члены структур  $ab$  и  $ba$ .

*Определение 3.* Отношение  $\omega$  на множестве  $U^+$  назовем отношением синтагматической эквивалентности, если  $(a, b) \in \omega$  тогда и только тогда, когда существует цепочка  $l \in U^+$  такая, что  $(l, a) \in \Delta$ ,  $(l, b) \in \Delta$ ,  $(l, ab) \in \Delta$  и  $(l, ba) \in \Delta$ .

Если  $(a_i, a_{i+1}) \in \omega$  ( $i = \overline{1, n-1}$ ), то цепочку  $a_1, a_2 \dots a_n$  назовем композицией цепочек  $a_1, a_2, \dots, a_n$ .

*Определение 4.* 1) Слово  $a$  входного словаря назовем сообщением, если существует цепочка  $b \in U^+$  такая, что  $(a, b) \in \delta$ .

2) Если  $a$  и  $b$  сообщения, то цепочки  $ab$  и  $ba$  такие, что  $(a, b) \in \delta$  или  $(a, b) \in \omega$ , будем называть сообщениями. Если число слов в  $ab$  и  $ba$  больше двух, то эти цепочки назовем проективными сообщениями.

3) Цепочку, являющуюся перестановкой слов проективного сообщения, будем называть непроективным сообщением, если в ней имеется хотя бы одна подцепочка вида  $abcd$  такая, что или  $ac$  и  $bd$  — синтагматические структуры, или одна из этих цепочек — синтагматическая структура, а другая — композиция.

4) Перестановку слов проективного сообщения назовем слабопроективным сообщением, если она не является непроективным сообщением

\* С. Ф. Липницкий, В. С. Яковишин // Весті АН БССР. Сер. фіз.-тэхн. навук. 1987. № 4.

и имеется хотя бы одна подцепочка вида  $abc$  или  $cba$  такая, что  $(b, a) \in \delta$ ,  $a(a, c) \in \delta$  или  $(a, c) \in \omega$ .

**Определение 5.** 1) Синтагматическую структуру  $abc$  назовем структурой с гнездованием, если: или  $(a, bc) \in \delta$  и  $(c, b) \in \delta$ , или  $(c, ab) \in \delta$  и  $(a, b) \in \delta$ .

2) Композицию  $abc$  будем называть композицией с гнездованием, если: или  $(a, bc) \in \omega$  и  $(c, b) \in \delta$ , или  $(ab, c) \in \omega$  и  $(a, b) \in \delta$ .

3) Сообщение назовем сообщением с гнездованием, если в нем содержится хотя бы одна синтагматическая структура или композиция с гнездованием. В противном случае будем говорить о сообщении без гнездования.

**Теорема 1.** Для всякого сообщения существует перестановка его слов, являющаяся проективным сообщением без гнездования.

Действительно, если сообщение непроективно или слабопроективно, то перестановка его слов, являющаяся проективным сообщением, существует в силу определения 4. Если же сообщение проективно, но для некоторой его подцепочки  $abc$   $(a, bc) \in \delta$ ,  $a(c, b) \in \delta$ , то транспозицией  $b$  и  $c$  получим цепочку  $acb$  без гнездования. Аналогично поступают и в других случаях гнездования.

**Теорема 2.** Если  $a_1a_2 \dots a_m, b_nb_{n-1} \dots b_1$  — проективные подцепочки без гнездования непроективного сообщения  $a_1a_2 \dots a_mb_nb_{n-1} \dots b_1$  такие, что для любых синтагм  $a_ia_j, b_lb_l$  ( $1 \leq i < j, j \leq m; 1 \leq l < k, k \leq n$ )  $(a_i, a_j) \in \delta, (b_l, b_k) \in \delta$ , то существуют подцепочки  $a_1a_2 \dots a_r$  ( $r \leq m$ ),  $b_sb_{s-1} \dots b_1$  ( $s \leq n$ ), конкатенация  $a_1a_2 \dots a_rb_sb_{s-1} \dots b_1$  которых является проективным сообщением, а подцепочка  $a_rb_s$  — синтагмой или композицией.

Существование указанной в условии теоремы 2 синтагмы или композиции следует из того, что цепочка  $a_1a_2 \dots a_mb_nb_{n-1} \dots b_1$  является сообщением, а существование проективной подцепочки — из проективности ее компонентов.

Теоремы 1, 2 позволяют реализовать синтаксический анализ непроективных и слабопроективных сообщений путем представления проективных подцепочек этих сообщений в виде синтагматических структур и композиций без гнездования и последовательного их исключения и восстановления. В силу теоремы 2 для анализа таких подцепочек может быть использован алгоритм из указанной работы. Пример анализа непроективного сообщения И ВОЛЧЬЮ ВАШУ Я ДАВНО НАТУРУ ЗНАЮ (И. А. Крылов) приведен в таблице.

На первом этапе анализ проводится в соответствии с алгоритмом из

Пример синтаксического анализа непроективного сообщения					
Этапы синтаксического анализа					
1	2	3	4	5	6
и волчью вашу я давно натуру знаю	и я давно натуру знаю	вашу и я давно натуру знаю	и я давно вашу натуру знаю	волчью и я давно вашу натуру знаю	и я давно вашу волчью натуру знаю

нашей работы\*. В результате получим совокупность проективных подцепочек (в нашем случае их четыре). Если исходное сообщение проективно, то этот этап анализа является единственным.

На втором этапе первая и вторая подцепочки упорядочиваются согласно условиям теоремы 2 и ищутся синтагмы\*. Если синтагма не найдена, то вторая подцепочка удаляется. Далее процедура повторяется.

На последующих этапах после получения единственной проективной подцепочки поочередно включаются удаленные подцепочки. После каждого включения осуществляется переупорядочение слов сообщения для получения подцепочек без гнездования. Завершается процесс анализа после получения единственной синтагматической структуры.

**Алгоритм синтаксического анализа.** На входе алгоритма — непроективное или слабoproективное сообщение  $a_1 a_2 \dots a_n$  ( $n \geq 2$ ), состоящее из проективных подцепочек  $a_i$  ( $i = \overline{1, n}$ ), полученных в результате синтаксического анализа в соответствии с приведенным алгоритмом\*, на выходе — синтагматическая структура или композиция. Алгоритм состоит из следующих шагов.

1.  $i := 1, j := 2, a := a_i, b := a_j, C = \emptyset$ .
2. Упорядочить цепочку  $ab$  (в соответствии с достаточными условиями проективности из теоремы 2).
3. Применить для анализа цепочки  $ab$  алгоритм\*. При успешном завершении процедуры анализа перейти к п. 4, в противном случае — к п. 5.
4.  $a := ab$ . Если  $j < n$ , то  $j := j + 1, b := a_j$ , перейти к п. 2, иначе — к п. 6.
5.  $C := C \cup \{a_j\}, j := j + 1, b := a_j$ , перейти к п. 2.
6.  $i := 2$ .
7. Если  $C = \emptyset$ , то упорядочить  $a$ . КОНЕЦ.
8. Если  $a_i \in C$ , то  $b := a_i$ , перейти к п. 9, иначе — к п. 10.
9. Упорядочить  $ab$  и применить алгоритм из указанной работы С. Ф. Липницкого, В. С. Яковишина. При успешном завершении процедуры анализа  $C := C \setminus \{a_i\}, a := ab$ , перейти к п. 7. В противном случае перейти к п. 10.
10. Если  $i < n$ , то  $i := i + 1$ , перейти к п. 8, иначе — к п. 6.

Очевидно, что алгоритм заканчивает свою работу не более чем за  $n-1$  проход шагов 2—5 и  $(n-2)$  — кратное выполнение шагов 6—9.

Поступила в редакцию 23.05.88.

УДК 681.3.06

К. А. ЗУБОВИЧ

## ПОСТРОЕНИЕ ПРЕОБРАЗОВАТЕЛЕЙ ПРЕДИКАТОВ И ГЕНЕРАЦИЯ МИНИМАЛЬНЫХ РЕШЕНИЙ ЛИНЕЙНЫХ ФУНКЦИОНАЛЬНЫХ УРАВНЕНИЙ В СИСТЕМЕ ФУНКЦИОНАЛЬНОГО ПРОГРАММИРОВАНИЯ

Одним из свойств широко обсуждаемых в настоящее время систем функционального программирования (СФП) [1] является упрощение процесса доказательства правильности программ [2—5]. Однако существует необходимость в разработке программных средств поддержки этого процесса [6]. В данной статье предлагается математическое обоснование и описывается программная реализация одного из этапов доказательства правильности программ в СФП — построения минимальных решений функциональных уравнений [2, 4, 6].

**Определение 1.** Функциональным уравнением в системе функционального программирования называется выражение вида  $X_i = E(X_i)$ , где  $X_i$  —

\* С. Ф. Липницкий, В. С. Яковишин // Весці АН БССР. Сер. фіз.-тэхн. навук. 1987. № 4.