Серия «Математика»
2011. Т. 4, № 4. С. 27—38

Онлайн-доступ к журналу:
http://isu.ru/izvestia

И З В Е С Т И Я
Иркутского государственного университета

УДК 510.5

Теория списков и Σ-определимость *

А. А. Гаврюшкина
Иркутский государственный университет

Аннотация. Списочной алгеброй над некоторым множеством S называется двусортная модель, основное множество которой состоит из множества S и списочной надстройки I_S над множеством S — множества списков элементов из $S \cup I_S$, снабженная естественными отношениями и операциями на списках (отношение принадлежности элемента списку, отношение подсписка, операции взятия головы и хвоста списка, операция присоединения элемента к списку). В данной работе показано, что операции в списочных алгебрах, которые могут быть заданы рекурсивно как по длине так и по глубине списка, являются Σ-определимыми.

Ключевые слова: теория списков; Σ-определимость; теорема о рекурсии.

1. Введение

Свойство упорядоченности присуще многим объектам информационного характера. Однако в классической логике свойство упорядоченности отсутствует, что затрудняет логическое описание упомянутых объектов. В качестве решения этой проблемы С.С. Гончаровым, Ю.Л. Ершовым и А.В. Свириденко [1, 2, 8] была введена теория списочных надстроек GES – теория, в которой объектами являются упорядоченные списки других объектов.

Данные качества теории GES предоставляют большие возможности для использования этой теории в задачах обработки данных и знаний в распределенных информационных пространствах. В частности, использование теории GES в методе объектно-ориентированных проекций [5, 6] вместо дескриптивных логик в значительной степени усиливает выразительность формализма. К сожалению, в связи с недостаточной изученностью теории GES дальнейшие исследования в данном

* Работа выполнена при финансовой поддержке Федеральной целевой программы «Научные и научно-педагогические кадры инновационной России» на 2009-2013 гг., государственный контракт № 16.740.11.0574 от 30 мая 2011 г.

направлении затруднены. Эта проблема послужила одним из основных мотивов для проведения данных исследований.

Таким образом, в данной статье мы продолжаем исследования теории списочных надстроек. Здесь мы докажем существенное свойство функций, задаваемых рекурсивно, быть Σ-определимыми в списочных алгебрах. Понятие Σ-определимости служит аналогом вычислимости на объектах, не являющихся натуральными числами. Таким образом, Σ-определимые функции в списочных надстройках являются вычислимыми функциями.

2. Списочная алгебра

Мы хотим рассмотреть конечные списки (упорядоченные цепочки), построенные из элементов некоторого множества S. Такие списки могут быть нелинейными, то есть среди элементов списка могут быть как элементы множества S так и сами списки. Кроме того, мы введем некоторые отношения и операции на этих списках.

Итак, пусть S — некоторое не более чем счетное множество. Рассмотрим минимальное множество слов I_S в алфавите $S \cup \{(,\ ),\ '','' \}$, где $S \cap \{(,\ ),\ '','' \} = \emptyset$, удовлетворяющее следующим условиям:

1) $() \in I_S$;

2) если $a \in S \cup I_S$, то $(a) \in I_S$.

3) если $a \in S \cup I_S$ и $(a_1, \dots, a_n) \in I_S$ для некоторых $a_1, \dots, a_n \in S \cup I_S$, то $(a, a_1, \dots, a_n) \in I_S$.

Легко видеть, что любой элемент полученного множества I_S либо является *пустым списком* () (будем обозначать его также через nil), либо представим единственным образом в виде $(a_1, \dots, a_n)$ для некоторых $a_1, \dots, a_n \in S \cup I_S$. То есть множество I_S образует множество конечных списков, построенных на основе элементов (праэлементов) из множества S, будем называть I_S *стандартной списочной надстройкой над* S.

Теперь мы хотим определить некоторую структура на множестве $D = S \cup I_S$. Определим отношения $\hat{\in}$ и $\hat{\subseteq}$:

$\hat{\in} \subseteq D \times I_S$ — отношение *принадлежности элемента списку*, где $a \,\hat{\in}\, l$, если $l = (a_1, \dots, a_n)$ для некоторых $a_1, \dots, a_n \in D$, $n \geq 1$ и $a = a_i$ для некоторого $1 \leq i \leq n$. Будем говорить, что a *элемент списка* l.

$\hat{\subseteq} \subseteq I_S \times I_S$ — отношение *подсписка*, где $l_1 \,\hat{\subseteq}\, l_2$, если либо $l_1 = ()$, либо $l_1 = (a_k, \dots, a_n)$ и $l_2 = (a_1, \dots, a_n)$ для некоторых $a_1, \dots, a_n \in D$ и $1 \leq k \leq n$. Будем говорить, что l_1 *подсписок* l_2.

Далее определим частичные операции на D:

$cons : D \times I_S \to I_S$ — операция *пополнения* списка новым элементом, где $cons(a, ()) = (a)$ и $cons(a, (a_1, \ldots, a_n)) = (a, a_1, \ldots, a_n)$ для $n \geq 1$. Заметим, что данная операция порождает множество D из множества $S \cup \{()\}$.

$head : I_S \to D$ — операция *взятия первого элемента* списка, где $head(()) = ()$ и $head((a_1, \ldots, a_n)) = a_1$ для $n \geq 1$.

$tail : I_S \to I_S$ — операция *взятия максимального собственного подсписка (хвоста)* списка, где

$$tail(()) = (), tail((a)) = () \text{ и } tail((a, a_1, \ldots, a_n)) = (a_1, \ldots, a_n)$$

для $n \geq 1$.

Таким образом, мы задали некоторую структуру с основным множеством D отношениями $\hat{\in}$ и $\hat{\subseteq}$, частичными операциями $cons$, $head$ и $tail$ и константой nil. Такую структуры будем называть *стандартной списочной алгеброй над множеством S*.

Так как операции определены частично, то со списочной алгеброй невозможно работать как с алгебраической системой в классическом понимании. В следующем параграфе будет введено понятие двусортной сигнатуры и двусортной модели, с помощью этих понятий мы сможем сформулировать и доказать некоторые полезные свойства списочных алгебр.

3. Двусортная модель

Определения классических понятий математической логики можно найти, например, в [4].

Нам понадобится некоторое расширенное определение модели (алгебраической системы), а именно двусортной модели с частичными операциями. Под двусортностью модели подразумевается, что элементы основного множества модели разделены на 2 сорта, что выражено разделением основного множества на два непересекающихся подмножества. Операции, отношения и константы в таких моделях также имеют определенные сорта (то есть операция может быть применена только к элементам определенного сорта), за счет этого в двусортных моделях могут появляться частично заданные операции.

Перейдем к определению двусортной модели. Для обозначения сортов будем использовать буквы α и β, а также сочетание $\alpha\beta$ для обозначения произвольного сорта. Определим двусортную сигнатуру. Пусть $\Sigma = (P \cup F \cup C, \nu)$ — сигнатура с множеством предикатных символов

P, множеством функциональных символов F, множеством константных символов C и функцией местности ν. Для задания двусортной сигнатуры нам понадобится функция

$$\mu : P \cup F \cup C \to \bigcup_{n \in \omega} \{\alpha, \beta, \alpha\beta\}^n$$

определяющая сорта сигнатурных символов. Эта функция должна удовлетворять следующим условиям:

- $\mu(f) \in \{\alpha, \beta, \alpha\beta\}^{\nu(f)+1}$ для $f \in F$,
- $\mu(p) \in \{\alpha, \beta, \alpha\beta\}^{\nu(p)}$ для $p \in P$,
- $\mu(c) \in \{\alpha, \beta, \alpha\beta\}$ для $c \in C$.

Тройку $(P \cup F \cup C, \nu, \mu)$ будем называть *двусортной сигнатурой* и обозначать $\Sigma_{\alpha,\beta}$.

Двусортной моделью двусортной сигнатуры $\Sigma_{\alpha,\beta}$ назовем тройку: $\mathcal{A} = (A, B, \Sigma^{\mathcal{A}}_{\alpha,\beta})$, где множества A и B не пересекаются, а $\Sigma^{\mathcal{A}}_{\alpha,\beta}$ интерпретация сигнатурных символов отношениями, операциями и элементами основного двусортного множества $A \cup B$.

Поясним, что мы подразумеваем под интерпретацией сигнатурных символов двусортной сигнатуры. Будем писать: *x имеет сорт α*, если $x \in A$; *x имеет сорт β*, если $x \in B$; и *x имеет сорт $\alpha\beta$*, если $x \in A \cup B$. Предикатные, функциональные и константные символы интерпретируются отношениями, частичными операциями и элементами множества $A \cup B$, соответственно, местность которых определяет функция ν. Кроме того, функция μ определяет сорта отношений, операций и констант:

– если $p \in P$ и $\mu(p) = (\alpha_1, \dots, \alpha_{\nu(p)})$, то $p^{\mathcal{A}}$ есть $\nu(p)$-местное отношение на $A \cup B$, такое, что если $(x_1, \dots, x_{\nu(p)}) \in p^{\mathcal{A}}$, то x_i имеет сорт α_i для $1 \leq i \leq \nu(p)$;
– если $f \in F$, $\mu(f) = (\alpha_1, \dots, \alpha_{\nu(f)}, \alpha_{\nu(f)+1})$ и x_i имеет сорт α_i для $1 \leq i \leq \nu(f)$, тогда значение $f^{\mathcal{A}}(x_1, \dots, x_{\nu(f)})$ определено и имеет сорт $\alpha_{\nu(f)+1}$;
– если $c \in C$, то $c^{\mathcal{A}}$ имеет сорт $\mu(c)$.

Примером двусортной модели является стандартная списочная алгебра. Пусть буквы ι и σ обозначают сорта элементов из множеств I_S и S, соответсвенно. Пусть $\Sigma_{\iota,\sigma} = (\hat{\in}^2, \hat{\subseteq}^2, cons^2, head^1, tail^1, nil)$ — двусортная сигнатура с функцией μ, заданной таблицей:

x	$\hat{\in}$	$\hat{\subseteq}$	$cons$	$head$	$tail$	nil
$\mu(x)$	$(\iota\sigma, \iota)$	(ι, ι)	$(\iota\sigma, \iota, \iota)$	$(\iota, \iota\sigma)$	(ι, ι)	ι

Тогда стандартная списочная алгебра $\mathcal{L} = (S, I_S, \Sigma_{\iota,\sigma})$ над множеством S является двусортной моделью сигнатуры $\Sigma_{\iota,\sigma}$.

Далее отношение принадлежности элементу списку и отношение подсписка мы будем обозначать $\in$ и $\subseteq$, опуская шляпки. Также для обозначения стандартной списочной алгебры над множеством S будем использовать более наглядную запись: $\mathcal{L} = (S, I_S, \in, \subseteq, cons, head, tail, nil)$.

4. Формулы

Определение термов и формул двусортной сигнатуры ничем не отличается от классического определения формул. Однако при определении семантики в связи с тем, что значение некоторых термов может оказаться неопределенным, необходимо внести некоторые уточнения в понятие истинности формулы на модели.

Будем говорить, что атомарная формула P *истинна на двусортной модели* $\mathcal{A}$ *при некотором означивании*, если все термы, которые встречаются в формуле P, определены при данном означивании и формула P истинна на модели $\mathcal{A}$ при данном означивании в классическом смысле. В противном случае атомарная формула P ложна. Определение истинности произвольной формулы дается как обычно.

Приведем определения Δ_0- и Σ-формул. Введем сокращения:

$$\exists x \in y\, \Phi \leftrightharpoons \exists x (x \in y \,\&\, \Phi)$$
$$\forall x \in y\, \Phi \leftrightharpoons \forall x (x \in y \to \Phi)$$

Определение 1. Δ_0-*формулами сигнатуры*

$$\Sigma_{\iota,\sigma} = \{\in^2, \subseteq^2, cons^2, head^1, tail^1, nil\}$$

являются:

– *атомарные формулы сигнатуры* $\Sigma_{\iota,\sigma}$*;*
– *если* φ *и* ψ *—* Δ_0*-формулы, то* $\neg\varphi$, $(\varphi \,\&\, \psi)$, $(\varphi \vee \psi)$ *и* $(\varphi \to \psi)$ *—* Δ_0*-формулы;*
– *если* φ *—* Δ_0*-формула, то* $\forall x \in y\, \varphi$ *и* $\exists x \in y\, \varphi$ *—* Δ_0*-формулы.*

Определение 2. Σ-*формулами сигнатуры* $\Sigma_{\iota,\sigma}$ *являются*

– *все* Δ_0*-формулы сигнатуры* $\Sigma_{\iota,\sigma}$*;*
– *если* φ *и* ψ *—* Σ*-формулы, то* $(\varphi \,\&\, \psi)$, $(\varphi \vee \psi)$ *—* Σ*-формулы;*
– *если* φ *—* Σ*-формула, то* $\forall x \in y\, \varphi$ *и* $\exists x \in y\, \varphi$ *—* Σ*-формулы;*
– *если* φ *—* Σ*-формула, то* $\exists x\, \varphi$ *—* Σ*-формула.*

Подробные сведения о понятиях Σ-формулы и Σ-определимости можно найти в [3, 7].

5. Теорема о рекурсии

В [1] была сформулирована и доказана следующая теорема:

Теорема 1. *Пусть функции $G(\overline{x})$ и $H(\overline{x}, u, v, w)$ Σ-определимы в некоторой списочной алгебре $\mathcal{L}$, тогда функция $F(\overline{x}, y)$, такая, что:*

$$\begin{aligned} &F(\overline{x}, ()) = G(\overline{x}) \\ &F(\overline{x}, cons(a, l)) = H(\overline{x}, a, l, F(\overline{x}, l)) \end{aligned}$$

также является Σ-определимой в $\mathcal{L}$.

Мы обобщили эту теорему:

Теорема 2. *Пусть функции $G_1(\overline{x})$, $G_2(\overline{x}, y)$ и $H(\overline{x}, y, u, v, w)$ Σ-определимы в некоторой стандартной списочной алгебре $\mathcal{L} = (S, I_S, \ldots)$, тогда функция $F(\overline{x}, y)$, такая, что:*

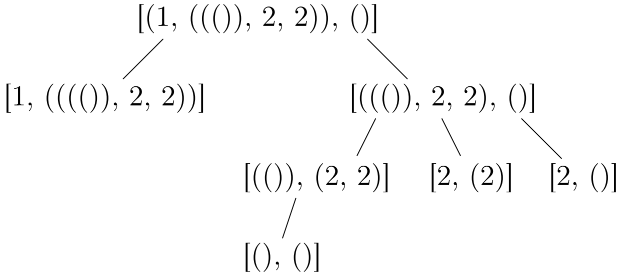
$$\begin{aligned} &F(\overline{x}, ()) = G_1(\overline{x}) \\ &F(\overline{x}, a) = G_2(\overline{x}, a) \text{ для } a \in S \\ &F(\overline{x}, cons(a, l)) = H(\overline{x}, a, l, F(\overline{x}, a), F(\overline{x}, l)) \end{aligned}$$

также является Σ-определимой в $\mathcal{L}$.

Для доказательства этой теоремы нам нужно ввести некоторые определения. *Упорядоченным деревом* называется дерево, в котором потомки любой вершины линейно упорядочены. *Деревом списка l* называется упорядоченное дерево T, каждая вершина которого есть двухэлементный список, корнем T является $(l, ())$ и для любой вершина v дерева T выполнены условия:

- если $head(v) = (v_1, \ldots, v_n)$, то v имеет ровно n потомков $u_1 < \ldots < u_n$, таких, что $u_i = (v_i, w_i)$ и $w_i = (v_{i+1}, \ldots, v_n)$ для $1 \leq i \leq n$ ($w_i = ()$ при $i = n$);
- если $head(v) = ()$ или $head(v) \in S$, то v не имеет потомков.

Рассмотрим пример дерева списка. Пусть $l = (1, ((()), 2, 2))$, на рисунке изображено дерево списка l, в котором потомки вершин указаны в порядке возрастания, для удобства внешние скобки изображены квадратными:



Будем обозначать через $\leq_{lex}$ лексикографический порядок на ветвях дерева списка, индуцированный порядком на потомках.

Каждую ветвь в дереве списка можно рассматривать как список, в котором первым элементом является корень, а каждый последующий элемент — потомок предыдущего. Мы также можем сравнивать с помощью отношения $\leq_{lex}$ списки, которым соответствуют ветви дерева списка.

Дерево списка, изображенное на рисунке, содержит 4 ветви, упорядоченные в порядке возрастания следующим образом:

$([(1,((()),2,2)),()],\ [1,(((()),2,2))])$

$([(1,((()),2,2)),()],\ [((()),2,2),()],\ [(()),(2,2)],\ [(),()])$

$([(1,((()),2,2)),()],\ [((()),2,2),()],\ [2,(2)])$

$([(1,((()),2,2)),()],\ [((()),2,2),()],\ [2,()])$

Предложение 1. *Пусть p_1 и p_2 — ветви дерева списка.*

1) Если $p_1 <_{lex} p_2$, то

$$p_1 = (x_1, \ldots, x_n, (y_1, z_1), \ldots), p_2 = (x_1, \ldots, x_n, (y_2, z_2), \ldots)$$

и $z_2 \subsetneq z_1$, для некоторых $x_1, \ldots, x_n, y_1, z_1, y_2, z_2, \ldots \in D$.

2) Если p_2 непосредственно следует за p_1 относительно $\leq_{lex}$, то

$$p_1 = (x_1, \ldots, x_n, (y_1, z_1), \ldots), p_2 = (x_1, \ldots, x_n, (y_2, z_2), \ldots)$$

и $z_1 = cons(y_2, z_2)$, для некоторых $x_1, \ldots, x_n, y_1, z_1, y_2, z_2, \ldots \in D$.

Доказательство. Следует из определения порядка $\leq_{lex}$ на ветвях и определения дерева списка. □

Заметим также, что:

Замечание 1. Для любых двух вершин (x_1, y_1) и (x_2, y_2) принадлежащих одной ветви дерева списка $x_1 \neq x_2$.

Доказательство. При переходе от вершины предка к вершине потомку длина первого элемента вершины (как слова) строго уменьшается. □

Списком ветвей дерева списка l называется список ll, элементами которого являются все ветви дерева списка, расположенные в порядке возрастания (относительно $\leq_{lex}$). Пусть ветвь $p = [(r_1, s_1), \ldots, (r_n, s_n)]$ является элементом списка ll, добавим в элементы этой ветви дополнительные элементы $a_1, b_1, \ldots, a_n, b_n \in D$, таким образом, что ветвь p преобразуется в ветвь $\hat{p} = [(r_1, a_1, s_1, b_1), \ldots, (r_n, a_n, s_n, b_n)]$. Заменив в ll все ветви на расширенные ветви получим *список $\hat{ll}$ расширенных ветвей дерева списка l* (порядок на расширенных ветвях остается прежний).

Пусть F — функция, определенная в теореме 2, пусть даны $\overline{x} \in D^n$ и список $l \in L_S$. Список $\hat{l}l$ расширенных ветвей дерева списка l назовем *F-списком расширенных ветвей дерева списка l для переменных $\bar{x}$*, если он удовлетворяет следующим условиям. Если

$$\hat{p}_0 = [(r_1^0, a_1^0, s_1^0, b_1^0), \ldots, (r_k^0, a_k^0, s_k^0, b_k^0)]$$

и

$$\hat{p}_1 = [(r_1^1, a_1^1, s_1^1, b_1^1), \ldots, (r_m^1, a_m^1, s_m^1, b_m^1)]$$

две ветви списка $\hat{l}l$, такие, что $\hat{p}_1$ непосредственно следует за $\hat{p}_0$, то:

1) если $s_i^0 = ()$, тогда $b_i^0 = G_1(\overline{x})$;

2) если $r_i^0 = ()$, то $a_i^0 = G_1(\overline{x})$;

3) если $r_i^0 = a$, где $a \in S$, то $a_i^0 = G_2(\overline{x}, a)$;

4) если $r_i^0 \in I_S$ и $r_i^0 \neq ()$, то

$$a_i^0 = H(\overline{x}, r_{i+1}^0, s_{i+1}^0, a_{i+1}^0, b_{i+1}^0) = F(\overline{x}, cons(r_{i+1}^0, s_{i+1}^0));$$

5) если $s_i^0 \neq ()$, $s_i^0 \neq s_i^1$ и $r_j^0 = r_j^1$, $s_j^0 = s_j^1$ для $j < i$, тогда $b_i^0 = a_{i-1}^1$;

6) Если $s_i^0 \neq ()$ и $r_j^0 = r_j^1$ и $s_j^0 = s_j^1$ для $j \leq i$, тогда $b_i^0 = b_i^1$.

Если ветвь p_0 последняя в списке $\hat{l}l$, то для нее должны быть выполнены условия 1 – 4.

Предложение 2. *Для любой расширенной ветви*

$$\hat{p} = [(r_1, a_1, s_1, b_1), \ldots, (r_n, a_n, s_n, b_n)]$$

F-списка $\hat{l}l$ расширенных ветвей дерева списка l для переменных $\bar{x}$ выполнено:

1) $a_i = F(\overline{x}, r_i)$ при $i = n$,

2) $a_i = F(\overline{x}, cons(r_{i+1}, s_{i+1}))$ при $1 \leq i < n$,

3) $b_i = F(\overline{x}, s_i)$ при $1 \leq i \leq n$.

Доказательство. Предложение доказывается индукцией по количеству ветвей списка $\hat{l}l$ и по их длине, используя пункт 2 предложения 1. □

Следствие 1. *Если $\hat{l}l$ — F-список расширенных ветвей списка l для переменных $\bar{x}$ и $\hat{p} = [(r_1, a_1, s_1, b_1), \ldots, (r_n, a_n, s_n, b_n)]$ — первый элемент списка $\hat{l}l$, то $a_1 = F(\overline{x}, l)$.*

Следующее предложение завершает доказательство теоремы о рекурсии.

Предложение 3. *Для любой функции F, удовлетворяющей условиям теоремы 2, существует Σ-формула $EPLists_F(\bar{x}, \hat{ll}, l)$, истинная в $\mathcal{L}$ тогда и только тогда, когда $\hat{ll}$ — F-список расширенных ветвей дерева списка l для переменных $\bar{x}$.*

Доказательство. Прежде всего заметим, что формула $\forall l' \subseteq l\, \Phi(\bar{x}, l', l)$ является Σ-формулой для любой Σ-формулы $\Phi(\bar{x}, l', l)$. Это следует из того, что список всех подсписков списка l определим Δ_0 формулой:

$$nil \in x \,\&\, \forall y \in x\, (y \subseteq l \,\&\, (\neg y = l \to \exists z \in l\, cons(y, z) \in x))$$

Несложно видеть, что следующие отношения определимы Σ-формулами:

– отношения $\mathbf{n}(x)$ (для $n \geq 1$), истинные тогда и только тогда, когда x — список из n элементов;
– отношение $Path(p, l)$, истинное, если l — непустой список, а p — ветвь (рассматриваемая как список) дерева списка l.

Рассмотрим функцию $unext : I_S \to I_S$, задаваемую рекурсивно по длине списка:

$$unext(()) = (),$$
$$unext(cons(a, l)) = cons((h(a), ht^2(a)), unext(l)), \text{ если } a \text{ — список},$$
$$unext(cons(a, l)) = cons(a, unext(l)), \text{ иначе.}$$

Так как отношение «не быть списком» определимо Δ_0-формулой $Void(x) \leftrightharpoons \forall y \in x\, \neg(y = y)$, то по теореме 1 эта функция определима некоторой Σ-формулой $\Phi_{unext}(\hat{p}, p)$. Используя эту формулу, несложно записать Σ-формулу, определяющую расширенные ветви.

Далее покажем, что отношение непосредственного следования на ветвях дерева списка является Σ-определимым. Рассмотрим функцию $\mathtt{conc} : I_S \times I_S \to I_S$, которая определяется следующим образом:

$$\mathtt{conc}(nil, y) = y,$$
$$\mathtt{conc}(cons(a, l), y) = cons(a, \mathtt{conc}(l, y))$$

Эта функция определима некоторой Σ-формулой $\Phi_{\mathtt{conc}}(x, y, z)$. Рассмотрим Σ-формулу $A(p_1, p_2, y_1, y_2)$, истинную тогда и только тогда, когда p_1 и p_2 — ветви некоторого дерева списка, y_1 и y_2 — минимальные подсписки p_1 и p_2, соответсвенно, такие, что для некоторого x выполнено $\mathtt{conc}(x, y_1) = p_1$ и $\mathtt{conc}(x, y_2) = p_2$ (миниимальность здесь равносильна тому, что $head(y_1) \neq head(y_2)$). Определим следующие два отношения:

$$SPath_l(p, p_1, l) \leftrightharpoons Path(p, l) \,\&\, p_1 \subseteq p \,\&$$

$$\forall x_1 \subseteq p_1 \, \forall x_2 \subseteq p_2[(t(x_1) = x_2 \,\&\, \neg(x_2 = nil)) \to h^2(x_2) = h^3(x_1)]^1$$

и

$$SPath_r(p, p_1, l) \;\leftrightharpoons\; Path(p, l) \,\&\, p_1 \subseteq p \,\&$$
$$\forall x \subseteq p_1 \, [(\neg x = p_1 \,\&\, \neg x = nil) \to hth(x) = nil]$$

Тогда отношение непосредственного следования определяется Σ-формулой:

$$p_1 \lhd p_2 \;\leftrightharpoons\; \exists l \, \exists y_1 \subseteq p_1 \, \exists y_2 \subseteq p_2[A(p_1, p_2, y_1, y_2) \,\&$$
$$SPath_l(p_1, y_1, l) \,\&\, SPath_r(p_2, y_2, l)]$$

С помощью формулы Φ_{unext} можно записать аналоги отношений A и $\lhd$ для расширенных ветвей (отношения A_e и $\lhd_e$). С помощью отношения $\lhd_e$ определяется список расширенных ветвей списка l, затем с помощью формул Φ_{G_1}, Φ_{G_2}, Φ_H, определяющих функции G_1, G_2, H, соответственно, и отношения A_e записываются формулы, обеспечивающие выполнение условий 1-6 определения F-списка (для выполнения условия 6 нужно воспользоваться замечанием 1).

$\square$

Теперь мы готовы к доказательству основной теоремы.

Доказательство теоремы 2. Согласно следствию 1 и предложению 3 функция $F(\bar{x}, y)$ определима Σ-формулой:

$$(\neg Void(y) \to \exists \hat{l}l[EPLists_F(\bar{x}, \hat{l}l, y) \,\&\, u = hth^2(\hat{l}l)]) \,\&$$
$$(y = nil \to \Phi_{G_1}(\bar{x}, u)) \,\&\, ((Void(y) \,\&\, \neg l = nil) \to \Phi_{G_2}(\bar{x}, l, u))$$

$\square$

6. Нестандартные списочные надстройки

В [2, 8] был также предложен аксиоматический подход к исследованию списочных надстроек. Ниже приведен список аксиом теории GES, в записях которых использованы обозначения: $\mathbf{r}, \mathbf{r}', \mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3$ для переменных сорта ι и e, e' для переменных сорта $\iota\sigma$.

AE: Аксиомы пустого списка

$$\neg\, e \in (), \quad () \subseteq \mathbf{r}$$

[1] здесь и далее используются сокращения: h для операции $head$ и t для операции $tail$, также при неоднократном применении этих операций опущены скобки и подряд повторяющиеся символы, количество которых указано в верхнем индексе. Например, $h^2t(x)$ означает $head(head(tail(x)))$.

AU: Аксиомы единственности

$$cons(e, \mathbf{r}) = cons(e', \mathbf{r}') \Rightarrow \mathbf{r} = \mathbf{r}' \wedge e = e'$$

AO: Аксиомы списочных операций:

$$\begin{aligned}
&head(()) = () \\
&tail(()) = () \\
&head(cons(e, \mathbf{r})) = e \\
&tail(cons(e, \mathbf{r})) = \mathbf{r} \\
&\mathbf{r} \neq () \to cons(head(\mathbf{r}), tail(\mathbf{r})) = \mathbf{r} \\
&e \in \mathbf{r} \ \wedge \ \mathbf{r} \subseteq \mathbf{r}' \to e \in \mathbf{r}'
\end{aligned}$$

AX: Аксиомы равнообъемности:

$$\begin{aligned}
\mathbf{r}_1 = \mathbf{r}_2 \Leftrightarrow (\forall \mathbf{r}_3 \subseteq \mathbf{r}_1)(\mathbf{r}_3 \subseteq \mathbf{r}_2 \ \wedge (\mathbf{r}_3 \neq \mathbf{r}_2 \vee \mathbf{r}_3 \neq \mathbf{r}_1) \to \\
\to (\exists e \in \mathbf{r}_1)(cons(e, \mathbf{r}_3) \subseteq \mathbf{r}_1 \wedge cons(e, \mathbf{r}_3) \subseteq \mathbf{r}_2)
\end{aligned}$$

AB: Аксиомы, определяющие $\in$:

$$e \in cons(e', \mathbf{r}) \Leftrightarrow (e = e' \vee e \in \mathbf{r})$$

AS: Аксиомы, определяющие $\subseteq$:

$$\mathbf{r} \subseteq cons(e, \mathbf{r}') \Leftrightarrow (\mathbf{r} \subseteq \mathbf{r}' \vee \mathbf{r} = cons(e, \mathbf{r}'))$$

AI: Аксиомы Σ–индукции (для всех Σ-формул F):

$$(F \,|^{x}_{()} \ \wedge \ \forall \mathbf{r}.\forall e.(F \,|^{x}_{\mathbf{r}} \Rightarrow F|^{x}_{cons(e,\mathbf{r})})) \Rightarrow \forall \mathbf{r}.F \,|^{x}_{\mathbf{r}}$$

AF: Аксиомы Σ-фундируемости (для всех Σ-формул F):

$$\forall \mathbf{r}.(\forall e \in \mathbf{r}.F \,|^{x}_{e} \Rightarrow F \,|^{x}_{\mathbf{r}}) \Rightarrow \forall e.F \,|^{x}_{e}$$

В таком случае *списочной алгеброй* называется любая двусортная модель сигнатуры $\Sigma_{\iota\sigma}$, в которой истинны все аксиомы GES (переменные в формулах, имеющие сорта, интерпретируются в моделях элементами соответствующих сортов). Теорема 2 верна в более общем случае:

Теорема 3. *Пусть функции $G_1(\overline{x})$, $G_2(\overline{x}, y)$ и $H(\overline{x}, y, u, v, w)$ Σ-определимы в некоторой списочной алгебре $\mathcal{L} = (S, I_S, \ldots)$, тогда функция $F(\overline{x}, y)$, такая, что:*

$$\begin{aligned}
&F(\overline{x}, ()) = G_1(\overline{x}) \\
&F(\overline{x}, a) = G_2(\overline{x}, a) \ \text{для } a \in S \\
&F(\overline{x}, cons(a, l)) = H(\overline{x}, a, l, F(\overline{x}, a), F(\overline{x}, l))
\end{aligned}$$

также является Σ-определимой в $\mathcal{L}$.

Список литературы

1. Гончаров С. С. Замечание об аксиомах списочной надстройки GES / С. С. Гончаров // Логические вопросы теории типов данных. – Новосибирск, 1986. – С. 11–15. –(Вычислительные системы, вып. 114).
2. Гончаров С. С. Σ-программирование / С. С. Гончаров, Д. И. Свириденко // Логико-математические проблемы МОЗ. – Новосибирск, 1985. – С. 3—29. – (Вычислительные системы, вып. 107).
3. Ершов Ю. Л. Определимость и вычислимость / Ю. Л. Ершов // Сибирская школа алгебры и логики. – Новосибирск : Научная книга, 1996.
4. Ершов Ю. Л. Математическая логика / Ю. Л. Ершов, Е. А. Палютин. – СПб. : Лань, 2004.
5. Малых А.А, Манцивода А.В, Ульянов В.С. Логические архитектуры и объектно-ориентированный подход // Вестник НГУ. Сер. Математика, механика, информатика. – 2009. – Т.9, вып. 3. -– С. 64–85.
6. А.А. Малых, А.В. Манцивода. Объектно-ориентированная дескриптивная логика // Изв. Иркут. гос. ун-та. Сер. Математика.– 2011. – Т.4, № 1. – С. 57–72.
7. J. Barwise. Admissible Sets and Structures. Perspectives in Mathematical Logic, Springer-Verlag Berlin Heidelberg NewYork, 1975.
8. Ershov, Yu.L, Goncharov, S.S., Sviridenko, D.I. Semantic Programming. Information processing, Proc. IFIP 10th World Comput. Congress, Dublin, vol. 10. pp 1093-1100, 1986.

Alexandra A. Gavryushkina

The theory of Lists and Σ-definability

Abstract. We consider two-sorted structures (lists algebras) consisting of a basic set S and a set of lists I_S (lists are ordered collections of elements from $S \cup I_S$) with natural relations and operations such as membership relation, head and tail operations etc. and show that recursively definable functions are Σ-definable in the lists algebras. The recursion is on the length and depth of a list.

Keywords: theory of lists, Σ-definability, recursion theorem

Гаврюшкина Александра Анатольевна, Институт математики, экономики и информатики, Иркутский государственный университет, 664003, Иркутск, ул. К. Маркса, 1 тел.: (3952)242210 (gavryushkina@gmail.com)

Alexandra A. Gavryushkina, Irkutsk State University, 1, K. Marks St., Irkutsk, 664003 Phone: (3952)242210 (gavryushkina@gmail.com)