

1. Рассмотрим следующую задачу: пусть $f, g: D^{n+1} \rightarrow D$ и $h: D \rightarrow D$ - вычислимые функции (D - множество символьных выражений с правильной скобочной структурой); заданы программы для функций g и h на некотором языке L_1 и соотношение $f(g(v, \bar{x}), \bar{x}) = h(v)$, на этом же языке. Требуется построить программу для функции f на некотором языке L_2 . Это одна из возможных формулировок обратной задачи. Одним из методов ее решения является преобразование программ.

В данной работе в качестве языка L_1 использовался модифицированный вариант "ограниченного Рефала", а в качестве языка L_2 и языка для написания самих алгоритмов преобразования - Рефал-2.

2. В контексте Рефала первые результаты в указанной области были получены в 1972 году [9]. Задача решалась в следующей постановке: задан предикат равенства $EQU(x, y)$, а $F(x)$ - функция, требуется найти такое значение x , что $EQU(F(x), c)$ - истинно, где x - переменная, а c - константа. Решение основывалось на Универсальном Решающем Алгоритме (УРА), позволяющем получить искомое значение переменной за конечное число шагов, если таковое существует. Первая реализация УРА на ЭВМ была выполнена С.А.Романенко, а в 1985 году С.М.Абрамовым была создана диалоговая УРА-система на ЕС ЭВМ, программы которой в модифицированном виде были использованы и для построения обратных функций в настоящей работе.

3. В докладе описываются два метода решения поставленной задачи: трансформационный и метод анализа конфигураций. Первый подход основан на преобразовании текстов описаний функций и применим для весьма ограниченного их класса (см., например [1]), конфигурационный анализ основывается на исследовании и преобразовании всего графа возможных историй вычисления функций [10, 11] и существенно расширяет область определения алгоритма обращения.

4. Суть трансформационного алгоритма (реализованного автором на ЭВМ) может кратко быть охарактеризована так: выполняя обобщенные шаги отождествления (прогонка) внутренних вызовов в левых частях предложений и подставляя соответствующие результаты, мы порождаем описания новых функций и пытаемся отождествить их с описаниями старых - в случае успешного отождествления алгоритм завершается, и объединенное описание становится самодостаточным. Узкая

направленность преобразований по сравнению с многочисленными правилами, используемыми в работах [2,3], позволила полностью автоматизировать этот процесс.

5. Граф конфигураций – это ориентированный граф, вершинами которого являются конфигурации – общие Рефал-выражения, представляющие обобщенные состояния (множества состояний) вычислительной системы, а все ребра помечены сужениями и присваиваниями, характеризующими соответствующие истории вычислений. Сужения имеют вид $V \rightarrow L$, где V – переменная, а L – выражение, допустимое в левых частях предложений ограниченного Рефала, присваивания имеют вид $E \leftarrow V$, где V – переменная, а E – общее выражение (в обозначениях работы [1]). В результате прогонки (обобщенного вычисления) в общем случае получается потенциально бесконечное дерево, однако, пользуясь предложенными в работе [10] методами обобщения, мы всегда можем получить конечный граф. Обобщения представляются в графе ребрами с присваиваниями.

6. Конфигурационный алгоритм обращения функции можно представить в виде последовательности шагов:

- построение конечного графа конфигурации для прямой функции;
- обращение графа прямой функции с учетом задания на обращение;
- преобразование "незаконных" сужений в "законные" присваивания на ребрах графа (порождение циклов при обращении конкатенации и элиминация вложенных вызовов функций);
- восстановление описания обратной функции.

Основная идея обращения графа прямой функции – это изменение направления всех ребер графа на противоположные с заменой всех сужений на присваивания и, наоборот, присваиваний на соответствующие сужения. Начальная конфигурация нового графа получается из левой части задания на обращение, а все остальные порождаются из нее с помощью сужений и присваиваний при обходе графа.

Для некоторого класса функций (например, функции сложения) этого достаточно, однако в общем случае возникают следующие вопросы:

А – порядок обхода ребер графа, т.е. порядок предложений в порождаемом описании функции;

Б – реализация неоднозначных сужений вида $V \rightarrow \dots e_1 \dots e_2 \dots$ (обращение конкатенации), получившихся из вполне однозначных присваиваний;

В - реализация сужений вида $v \rightarrow \dots \langle f, E \rangle \dots$, не допустимых в левых частях предложений.

Порядок обхода ребер (вопрос А) в большинстве случаев наследуется из прямой функции, но мы пока ограничиваемся классом функций с независимыми сужениями на ребрах, для которых указанный порядок несуществен. При обращении конкатенации (вопрос Б) в графе порождается цикл перебора для нахождения искомого разложения переменной. В качестве решения последней проблемы (вопрос В) предлагается преобразование соответствующего сужения в присваивание с заменой вызова функции на вызов обратной к ней. Таких методов оказывается достаточно, например, для построения обратных функций (по первому или второму аргументам) к функции Аккермана.

7. В заключение рассматриваются некоторые расширения языка, которые подсказала деятельность по обращению функций.

В соответствии с первым расширением в правых частях предложений допускается использование переменных, не встречавшихся в соответствующих левых частях. Содержательно, это дает возможность описывать (и порождать!) функции, результатом вычисления которых будут множества, представленные типовыми выражениями, т.е. появляются возможность нахождения обратных не только для инъективных функций.

Второе расширение носит более "синтаксический" характер и вводит в Рефал изображение степени, основанием которой может быть любое выражение, а показателем должно быть число или переменная символа, отождествляемая с числом. При этом, за счет усложнения встроеного алгоритма отождествления левых частей, повышается наглядность в представлении определенного класса функций, а кроме того мы получаем естественное для Рефала выражение длины. При некоторых ограничениях на основание, введение степени не противоречит системе эквивалентных преобразований, сформулированных для Рефала и лежащих в основе построения графа конфигураций.

8. Дальнейшая работа движется в направлении нахождения "неподвижной точки" как в языковой области (сведение всех трех языков: L_1 , L_2 и языка реализации к одному, например, Рефалу-4 [8]), так и в классе обрабатываемых функций.

Литература

1. Burstall R.M. Inductively Defined Functions. - LNCS, 185, 1985, pp.92-96.
2. Darlington J. An Experimental Program Transformation and Synthesis System. - Artificial Intelligence, Vol 6, 1981, pp.1-46.

3. **Feather M.S. A System for Assisting Program Transformation.- ACM Trans. on Prog. Lang. and Syst. 4(1), 1982, pp.1-20.**
4. **Jones N.D., Sestoft P., Sondergaard H. An Experiment in Partial Evaluation: The Generation of a Compiler Generator.- Sigplan Notices, 20(8), 1985, pp. 82-87.**
5. **Катленд Н. Вычислимость. Введение в теорию рекурсивных функций. М.; "Мир", 1983.**
6. **Климов А.В., Романенко С.А. Система программирования РЕФАЛ-2 для ЕС ЭЕМ. Описание входного языка. Препринт ИПМ им.М.В.Келдыша АН СССР, М.: 1987.**
7. **Романенко С.А. Генератор компиляторов, порожденный самоприменением специализатора, может иметь ясную и естественную структуру. - Препринт ИПМ им.М.В.Келдыша АН СССР, № 26 за 1987 год.**
8. **Романенко С.А. Рефал-4 - расширение Рефала-2, обеспечивающее представление результатов прогонки. - Препринт ИПМ им.М.В.Келдыша АН СССР, № 147 за 1987 год.**
9. **Турчин В.Ф. Эквивалентные преобразования рекурсивных функций, описанных на языке РЕФАЛ. - в Теория языков и методы построения систем программирования. Труды симпозиума. Киев-Алушта, 1972, с.31-42.**
10. **Turchin V. The Language Refal, the Theory of Compilation and Metasystem Analysis. - Technical Report No 18, Courant Institute of Mathematics, NY, 1980.**
11. **Turchin V. The Concept of a Supercompiler. - ACM Trans. on Prog. Lang. and Syst. 8(3), 1986, pp. 292-325.**

**СЕМИОТИЧЕСКИЕ АСПЕКТЫ
ФОРМАЛИЗАЦИИ
ИНТЕЛЛЕКТУАЛЬНОЙ ДЕЯТЕЛЬНОСТИ**

**Всесоюзная школа-семинар
«Боржоми-88»**

ТЕЗИСЫ ДОКЛАДОВ И СООБЩЕНИЙ

МОСКВА 1988

Всесоюзный институт научной и технической информации

ГКИТ СССР и АН СССР

Институт кибернетики АН Грузинской ССР

Вычислительный центр АН СССР

**Научный совет по проблеме "Искусственный интеллект" Отделения
информатики, вычислительной техники и автоматизации АН СССР**

**СЕМИОТИЧЕСКИЕ АСПЕКТЫ
ФОРМАЛИЗАЦИИ
ИНТЕЛЛЕКТУАЛЬНОЙ ДЕЯТЕЛЬНОСТИ**

**Всесоюзная школа-семинар
г.Боржоми, 22-30 апреля 1988 г.**

ТЕЗИСЫ ДОКЛАДОВ И СООБЩЕНИЙ



Москва 1988

Председатели Оргкомитета чл.-корр. АН ГССР Г.Д. х а р а т и ш в и л и
проф. П.В. Неостеров

Председатель Программного комитета академии Г.С. П о с п е л о в

Руководители секций

Проф. Д.А. П о с п е л о в , докт. филол.наук Вяч.В.И в а н о в

Составитель

к.с.и. В.К. Ф и н и

Редакторы

к.ф.-м.и. М.М. З а б е ж а й л о , Е.В. Р а х и л и н а ,
В.В. Р ы ж и н о в , к.ф.-м.и. Д.П. С и в о р ц о в ,
Е.Ф. Ф е б р и а н т о в а , М.В. Ф и л и п е н н о

Ответственный редактор

доц. Р.С. Г и я р е в о и я

© ВИНИТИ, 1988 г.

С Е К Ц И Я I

ЛОГИЧЕСКИЕ ПРОБЛЕМЫ И ПРОГРАММНЫЕ СРЕДСТВА
ПОСТРОЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

Руководитель

Д.А. П о с п е л о в

Ученый секретарь

В.Ф. Ф а б р и к а н т о в а

	Сдано в набор 15,03,88	
Подписано в печать 26,02,88		Т-02869
Формат 60x90 1/16	Печать офсетная	Бум. офс.
Усл.печ.л 26,5	Усл.кр.-отт. 26,62	Уч.-изд.л 22,81
Тир. 250 экз.	Зах. 2226	Цена 1р.55к.

Производственно-издательский комбинат ВИНТИ
140010, Люберцы 10, Московской обл.,
Октябрьский проспект, 403