

На правах рукописи

БУТАКОВ МИХАИЛ ИГОРЕВИЧ

ИНСТРУМЕНТАЛЬНОЕ СРЕДСТВО СИНТЕЗА И
ИСПОЛНЕНИЯ ТРАНСЛИРУЮЩИХ ПРОГРАММ НА
ОСНОВЕ ПОЗИТИВНО-ОБРАЗОВАННЫХ ФОРМУЛ

Специальность 05.13.11 – Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата технических наук

Иркутск – 2012

Работа выполнена в Учреждении Российской академии наук Институте динамики систем и теории управления Сибирского отделения Российской академии наук (ИДСТУ СО РАН).

Научный руководитель: кандидат физико-математических наук,
доцент
Курганский Виктор Иванович

Официальные оппоненты: доктор физико-математических наук,
профессор
Манцивода Андрей Валерьевич

кандидат технических наук
Черкашин Евгений Александрович

Ведущая организация: **Институт проблем управления
им. В.А. Трапезникова РАН
(г. Москва)**

Защита состоится **15 марта 2012 г. в 14:00** на заседании диссертационного совета Д 003.021.01 в ИДСТУ СО РАН по адресу: 664033, г. Иркутск, ул. Лермонтова, 134.

С диссертацией можно ознакомиться в библиотеке и на официальном сайте www.idstu.irk.ru ИДСТУ СО РАН.

Автореферат разослан **14 февраля 2012 г.**

Ученый секретарь
диссертационного совета
д.ф.-м.н.



А.А. Щеглова

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Диссертация посвящена применению исчисления позитивно-образованных формул (ПОФ) в прикладных программах с целью решения задачи трансляции для формальных языков второго и третьего класса по классификации Хомского.

Актуальность темы исследования. Для расширения функциональных возможностей современных прикладных программ часто приходится решать задачи трансляции: макроязык VBA в Microsoft Office, макроязык AutoLisp в AutoCad, предметно-ориентированный язык 1С: Предприятие в продуктах 1С и др.

Применение встроенных в прикладные программы средств трансляции и исполнения программного кода, формируемого пользователем, позволяет связывать воедино отдельные функциональные блоки на основе целостной формальной спецификации и синтезировать новые программы обработки данных на основе имеющихся подпрограмм.

Одним из подходов к решению указанных задач является представление формального языка и входной цепочки в виде логической теории первого порядка. Применение логических средств для решения задач трансляции позволяет:

- адаптировать средства интеллектуализации к терминологии и свойствам предметной области;
- сократить время и трудоемкость разработки прикладных программ со встроенными средствами трансляции;
- повысить качество компонентов прикладной программы за счет их синтеза в результате дедуктивного вывода из спецификаций компонентов и исходных данных;
- проверять непротиворечивость спецификаций, разработанных на основе известных алгоритмов.

В связи с этим развитие на основе логического подхода средств конструирования трансляторов, встраиваемых в прикладные программы, является актуальным.

При разработке встроенных систем трансляции на основе логического подхода огромное значение имеет компактное представление знаний и хорошая адаптация к решаемой задаче. ПОФ в сравнении с классическими логическими исчислениями обеспечивают более эффективные стратегии логического вывода и обладают рядом важных свойств, необходимых при разработке представленных систем: формулы языка имеют простую и регулярную структуру; позитивно-образованное представление для первопорядковых формул более компактно в сравнении с языком дизъюнктов; формулы языка сохраняют эвристическую структуру знания при выводе; ПОФ, за счет разработки дополнительных стратегий логического вывода, хорошо адаптируются к решаемой задаче.

Большая часть работы основана на результатах, полученных С.Н. Васильевым и его учениками в области логического вывода и приложений логического вывода в составе пакетов прикладных программ. Предпосылкой для исследования послужили работы Ершова А.П. по доказательному программированию. При решении части задач были использованы идеи дедуктивного и структурного синтеза программ, изложенные в работах З. Манна, Р. Уолдингера и Э.Х. Тыугу. Существенное влияние на результаты диссертации оказали работы Н. Хомского, А. Ахо и Дж. Ульмана, посвященные моделям и методам трансляции.

Объединение идей логического вывода и его приложений в управлении, классических методов трансляции, результатов доказательного программирования и дедуктивного синтеза программ позволило получить простой и универсальный метод автоматического построения транслирующих программ на основе позитивно-образованных формул.

Цель и задачи исследования. Цель работы – на основе ПОФ разработать инструментальное средство конструирования и применения транслирующих программ в составе прикладных программ для формальных языков второго и третьего классов по Хомскому.

Для достижения указанной цели были решены следующие задачи:

1. Разработать постановку и схему решения задачи трансляции в исчислении ПОФ;
2. Разработать алгоритмы построения распознающих ПОФ по регулярной правосторонней детерминированной грамматике и LL(1)-грамматике;
3. Разработать инструментальное средство для конструирования и применения в составе прикладных программ транслирующих ПОФ;
4. Применить разработанное инструментальное средство на практике.

Объектом исследования является теория и практика спецификации транслирующих программ при помощи логического языка первого порядка. Под транслирующей программой понимается программа, которая принимает на вход цепочку символов (исходный код), представленную на одном языке, и преобразует ее в выходную цепочку символов (объектный код) на другом языке.

Предметом исследования является метод автоматического синтеза транслирующих программ обработкой ее спецификации в виде ПОФ.

Методы исследования. В исследовании использовались методы математической логики, теории формальных языков, теории трансляции и синтеза программ, а также теории программирования.

Научная новизна. В диссертации впервые предложена постановка и схема решения задачи трансляции на основе доказательства существования и синтеза транслирующей программы в исчислении ПОФ. Для решения этой задачи разработана стратегия логического вывода τ . Использование ПОФ

для решения задач трансляции позволяет объединить решение задачи единым формализмом с другими знаниями в рамках интеллектуальных подсистем. Для применения схемы на практике разработаны транслирующие ПОФ и алгоритмы их построения по регулярным и LL(1)-грамматикам. Транслирующие ПОФ эффективнее других решений задач трансляции на основе логической теории первого порядка, так как они имеют более развитые средства управления логическим выводом и позволяют представить задачу трансляции более компактно. Схема решения и разработанные алгоритмы интегрированы в разработанное инструментальное средство для конструирования и применения транслирующих ПОФ. С помощью инструментального средства было получено новое решение задачи контроля диалога человека и объектной программы. Полученное решение упрощает решение задачи контроля диалога за счет представления требований к диалогу в виде двухуровневой иерархии формальных языков, каждый из которых задается транслирующей ПОФ.

Практическая значимость. Предложенное инструментальное средство конструирования транслирующих программ на основе ПОФ дает разработчикам возможность дополнить создаваемые прикладные программы подсистемами трансляции, визуального проектирования и отладки формальных языков. При этом процесс отладки формальных языков значительно упрощается за счет интерпретации вывода ПОФ в терминах вопросно-ответных процедур, что, в свою очередь, улучшает функциональные возможности и гибкость прикладных программ.

Разработанные алгоритмы преобразования регулярных и LL(1)-грамматик в ПОФ могут быть использованы для построения более сложных инструментов решения задач трансляции на основе формул первопорядковой логики.

Представление требований к диалогу человека и объектной программы в виде двухуровневой иерархии формальных языков может применяться для решения задач контроля диалога во многих прикладных диалоговых системах.

Разработанное инструментальное средство компонент синтеза транслирующих программ (КСТП), алгоритмы преобразования грамматик в ПОФ и представление требований к диалогу в виде ПОФ применены при решении двух практических задач: разработки учебных трансляторов и контроля диалога в диалоговой системе Терминал.

Достоверность результатов. Достоверность полученных в работе результатов подтверждена разработкой алгоритмов построения ПОФ по регулярным и LL(1)-грамматикам и доказательством эквивалентности грамматического разбора регулярных и LL(1)-языков процессу доказательства существования соответствующей транслирующей программы.

Эффективность полученных в работе результатов подтверждена опытом практической эксплуатации разработанного автором инструментального средства. Данное инструментальное средство применяется при решении учебных задач трансляции в ИМЭИ ИГУ в рамках курса «Языки программирования и методы трансляции» с 2007 г. по настоящее время. С помощью КСТП решена задача контроля диалога в рамках специализированной диалоговой системы таможенного контроля лесоматериалов. Диалоговая система контроля лесоматериалов рекомендована Федеральной таможенной службой России к экспериментальному опробованию в таможенных органах Сибирского таможенного управления и Дальневосточного таможенного управления¹.

Соответствие диссертации паспорту научной специальности. В диссертации представлена новая модель и схема решения задачи трансляции в исчислении ПОФ. Результаты работы могут быть использованы при проектировании прикладных программ и систем, включающих трансляцию формальных языков. Одним из аспектов использования является контроль допустимости цепочек входных воздействий в составе диалогового интерфейса прикладной программы. Таким образом, тематика диссертации соответствует пунктам 1, 7 области исследований специальности 05.13.11.

Апробация работы. Основные результаты работы были представлены на научно-теоретической конференции молодых ученых, посвященной 60-летию Великой Победы (Иркутск, 2005), на VII Всероссийском симпозиуме по прикладной и промышленной математике (Кисловодск, 2006), на VIII школе-семинаре молодых ученых «Математическое моделирование и информационные технологии: управление, искусственный интеллект, прикладное программное обеспечение, технологии программирования» (Иркутск, 2006), на IX школе-семинаре молодых ученых «Математическое моделирование и информационные технологии» (Иркутск, 2007), на XIV Байкальской Всероссийской конференции «Информационные и математические технологии в науке и управлении» (Иркутск, 2009), на III школе-семинаре молодых ученых «Инфокоммуникационные и вычислительные технологии и системы» (Улан-Удэ, 2010), на 4-й Всероссийской мультikonференции по проблемам управления (с. Дивноморское, 2011) и неоднократно на семинарах Института динамики систем и теории управления СО РАН и Кафедры информационных систем Института математики, экономики и информатики ГОУ ВПО ИГУ.

Результаты диссертации были получены в рамках проекта СО РАН «Интеллектуальные методы и инструментальные средства создания и анализа интегрированных распределенных информационно-аналитических и вычислительных систем для междисциплинарных исследований с

¹ О распространении системы электронного поштучного учета: Письмо ФТС России от 11 марта 2008 г. № 01-11/9055).

применением ГИС, GRID и Web-технологий» (№ гос. регистрации 01.2.00708582), 2007–2009 гг.

Публикации и личный вклад автора. По теме диссертации опубликовано 12 работ, которые включают 3 статьи [1–3] в журналах, рекомендованных ВАК для опубликования результатов диссертаций, 3 публикации [6, 11, 12] в трудах региональных и всероссийских конференций, 5 публикаций [4, 5, 7, 8, 10] в научных сборниках и журналах, 1 свидетельство об официальной регистрации программ в Федеральной службе по интеллектуальной собственности, патентам и товарным знакам [9].

Результаты главы 1 опубликованы в работах [1, 4, 5, 7]; результаты главы 2 опубликованы в [1, 2, 4, 5, 7, 8]; результаты главы 3 опубликованы в [1, 3, 6, 10, 12]; результаты главы 4 опубликованы в [2, 3, 8, 9, 11].

Все результаты, выносимые на защиту, получены автором лично. В работах [1, 3–5, 7] В.И. Курганскому принадлежат постановки исследуемых задач. В работах [2, 8] О.В. Курганской принадлежат результаты по диаграммному представлению ПОФ. В работе [9] Ю.Д. Королькову и В.И. Курганскому принадлежит решение вопросов, связанных с технологией, организацией и осуществлением учебного процесса.

Структура и объем диссертации. Диссертация состоит из введения, четырех глав, заключения, библиографии и двух приложений. Общий объем диссертации – 123 страницы. Библиография включает 103 наименования.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Во **введении** сформулированы цели и дана общая характеристика работы, поставлены задачи, отражена актуальность, практическая значимость и научная новизна исследования.

Первая глава посвящена постановке и схеме решения задачи трансляции. Задача решается доказательством существования и синтезом транслирующей программы на основе логического вывода в частном случае конструктивного фрагмента исчисления J позитивно-образованных формул определенного вида.

Исчисление J построено в языке ПОФ L . Язык L – полный язык первого порядка, формулы которого имеют древовидную структуру.

Алфавит языка L описывается множеством символов следующих видов:

- $x, y, z, \dots$ – символы, обозначающие переменные;
- $a, b, c, \dots$ – константы;
- $\geq, =, A, B, \dots$ – предикатные символы различной арности;
- $(,)$ – запятая, скобки для уточнения строения формул;
- $\forall, \exists$ – знаки для кванторов всеобщности и существования соответственно.

Атомы языка L имеют вид $R^k(t_1, \dots, t_k)$, где R^k – предикатный символ арности k , а каждое t_i ($i = 1, \dots, k$) – переменная или константа.

Конечное множество атомов, записанных через запятую, и пропозициональные константы **T** (истина) и **F** (ложь) являются конъюнктами в языке L .

Элементарные формулы языка L имеют вид:

- $\exists$ -формулы: $\exists X : A$;
- $\forall$ -формулы: $\forall X : A \exists F$,

где X – множество переменных, A – конъюнкт.

Если $\Phi = \{\Phi_1, \dots, \Phi_k\}$ является множеством $\forall$ -формул, то $\exists Y : B \Phi$ является $\exists$ -формулой. При $k \geq 2$ ветвление формул множества Φ понимается как конъюнктивное.

Если $\Psi = \{\Psi_1, \dots, \Psi_k\}$ является множеством $\exists$ -формул, то $\forall Y : B \Psi$ является $\forall$ -формулой. При $k \geq 2$ ветвление формул множества Ψ понимается как дизъюнктивное.

Далее записью $\forall T\{\Psi, \exists X : A \Phi\}$ будем обозначать формулы языка ПОФ². Другой способ записи формул языка L имеет вид

$$F = \forall : T \overset{def}{\left\{ \begin{array}{l} \exists X : A \{ \Phi \\ \Psi \end{array} \right.$$

где Ψ – множество $\exists$ -формул, принадлежащих языку L , Φ – множество ветвей вида

$$\forall Y : B \{ \exists C_i : Z_i \Psi_i \}_{i=1, \dots, k} \overset{def}{=} \forall Y : B \left\{ \begin{array}{l} \exists C_1 : Z_1 \{ \Psi_1 \\ \dots \\ \exists C_k : Z_k \{ \Psi_k \end{array} \right.$$

$\Psi_1, \dots, \Psi_k$ – множество $\forall$ -формул, принадлежащих языку L .

Исчисление J в языке L имеет единственное правило вывода ω , основой которого является понятие подстановки θ .

Любой непосредственный последователь $\forall Y : B$ базы $\exists X : A$ называется вопросом к базе $\exists X : A$. Говорят, что вопрос $\forall Y : B$ к базе $\exists X : A$ имеет ответ (ответную подстановку) θ тогда и только тогда, когда θ есть суть отображение $Y \rightarrow X$ и $B\theta \subseteq A$.

Пусть ПОФ F имеет вид $\forall T\{\Psi, \exists X : A \Phi\}$, а Φ содержит подформулу $\forall Y : B \{ \exists Z_i : C_i \Psi_i \}_{i=1, \dots, k}$, тогда результатом ωF применения правила вывода ω к вопросу $\forall Y : B$ с соответствующей ответной подстановкой θ будет формула $\omega F = \forall T\{\Psi, \{ \exists X \cup Z_i : A \cup C_i \theta \Phi \cup \Psi_i \theta \}_{i=1, \dots, k} \}$.

² Vassilyev S.N., Davydov A.V., Zherlov A.K. Intelligent control via new efficient logics // Proc. of the 17th IFAC World Congress. Seoul (Korea), 2008. P. 13713–13718.

Логический вывод ПОФ в исчислении J представляет собой цепочку эквивалентных преобразований исходной формулы F по правилу ω , заканчивающуюся формулой $\forall : T \exists : F$. Если ПОФ не содержит ветви $\exists : F$, то она невыводима.

Определим подмножество языка L ПОФ, пригодных для решения задачи трансляции. Под решением задачи трансляции будем понимать доказательство существования транслирующей программы (преобразующую заданную входную цепочку в выходную цепочку), синтез транслирующей программы и ее последующее исполнение. Транслирующая программа строится из заранее заданного набора шаблонов команд запуска транслирующих подпрограмм.

Выделим вид ПОФ, пригодных для решения задач трансляции. Для этого рассмотрим теорему о существовании транслирующей программы в исчислении предикатов первого порядка

$$\left(\left((\forall X_1 : (P_1(X_1) \rightarrow \exists Y_1 : R_1(X_1, Y_1))) \wedge \dots \right. \right. \\ \left. \left. \dots \wedge (\forall X_n : (P_n(X_n) \rightarrow \exists Y_n : R_n(X_n, Y_n))) \right) \rightarrow (\forall X : (P(X) \rightarrow \exists Y : R(X, Y))) \right), \quad (1)$$

где $\forall X_1 : (P_1(X_1) \rightarrow \exists Y_1 : R_1(X_1, Y_1))$, ..., $\forall X_n : (P_n(X_n) \rightarrow \exists Y_n : R_n(X_n, Y_n))$, $\forall X : (P(X) \rightarrow \exists Y : R(X, Y))$ – спецификации транслирующих подпрограмм G_1 , ..., G_n и транслирующей программы G соответственно. Формулы $P_1(X_1)$, ..., $P_n(X_n)$, $P(X)$ играют роль предусловий рассматриваемых программ, а формулы $R_1(X_1, Y_1)$, ..., $R_n(X_n, Y_n)$, $R(X, Y)$ – их постусловий.

Элементы $P_1(X_1)$, ..., $P_n(X_n)$, $P(X)$ и $R_1(X_1, Y_1)$, ..., $R_n(X_n, Y_n)$, $R(X, Y)$ являются конъюнктами, которые строятся из атомов, обозначающих возможность разбора элемента входной цепочки, успешный разбор элемента входной цепочки, разбор языковой конструкции, обозначенной нетерминальным символом грамматики, а также успешное завершение разбора.

На языке ПОФ теорема (1) имеет вид:

$$F = \forall : T \ U_B \{ U_Q \}, \quad (2)$$

где база ПОФ $U_B = \exists X : P(X)$, а множество вопросов $U_Q = Q_1, \dots, Q_n \forall Y : R(X, Y) \exists F$, $Q_i = \forall X_i : P_i(X_i) \exists Y_i : R_i(X_i, Y_i)$. Часть $\forall : T$ в начале формулы необходима для построения дизъюнктивного ветвления при доказательстве ПОФ.

Формулы вида (2) позволяют решать задачу трансляции. Далее такие формулы будем называть распознающими позитивно-образованными формулами.

Для решения задачи трансляции с помощью распознающих ПОФ примем ряд соглашений:

- при описании структуры цепочки используются нетерминальные и терминальные символы;
- разбор входной цепочки выполняется слева направо по одному символу;
- конец входной цепочки обозначен специальным терминальным символом;
- в общем случае каждому нетерминальному символу, т.е. конструкции формального языка, соответствует транслирующая подпрограмма. Транслирующая подпрограмма задает преобразование конструкции формального языка или диагностирует ошибку.

С учетом представленных соглашений дополним правило вывода ПОФ ω новой стратегией вывода τ .

Обозначим через ρ , ω_1 и ω_2 составные элементы стратегии вывода τ .

Элемент ρ стратегии τ заключается в разделении базы U_B позитивно-образованной формулы F на два конъюнкта: конъюнкт синтеза и конъюнкт вывода. Конъюнкт синтеза K_S составляют атомы, уже использованные при преобразовании по правилу ω формулы F , а конъюнкт вывода K_D – атомы, которые можно использовать в данный момент при применении правила ω . После каждого успешного применения правила ω конъюнкт синтеза пополняется новыми атомами из конъюнкта вывода, а в конъюнкт вывода копируется постусловие обрабатываемого вопроса. Формальные параметры постусловия при этом должны быть заменены переменными вывода в соответствии с ответной подстановкой θ .

Элемент ω_1 стратегии τ заключается в поиске подходящего вопроса для выполнения преобразования формулы F по правилу ω путем просмотра конъюнкта вывода слева направо. Глубина просмотра при проверке выполнимости вопроса определяется мощностью множества атомов, составляющих конъюнкт данного вопроса. Критерием применимости правила ω к данному вопросу является принадлежность конъюнкту вывода каждого атома предусловия вопроса, обработанного ответной подстановкой θ .

Элемент ω_2 стратегии τ заключается в пополнении конъюнкта вывода атомами постусловия, примененного в рамках правила ω вопроса, путем их приписывания к конъюнкту вывода только слева.

Невозможность применения элемента ω_1 стратегии τ означает недоказуемость распознающей ПОФ. Случай, когда конъюнкт вывода содержит F , означает, что распознающая позитивно-образованная формула успешно доказана.

По существу доказательство распознающей ПОФ (2) позволяет выполнить распознавание цепочек формального языка. Для синтеза транслирующей программы введем другой вид ПОФ – транслирующие позитивно-образованные формулы.

Транслирующие ПОФ отличаются от распознающих формул двумя моментами:

1. С каждым атомом начальной установки конъюнкта вывода K_D и каждым атомом постусловия каждого вопроса транслирующей формулы может быть связан нелогический элемент – шаблон команды запуска транслирующей подпрограммы;

2. Набор вопросов транслирующей формулы шире и содержит вопросы, соответствующие возможным ошибкам распознавания входной цепочки и дополнительным (не учтенным в распознающей ПОФ) условиям.

Так как входная цепочка входит в состав рассматриваемых ПОФ, то синтезированная программа представляет собой последовательность команд безусловного запуска транслирующих подпрограмм.

Команда запуска транслирующей подпрограммы включается в синтезированную программу после успешного шага доказательства позитивно-образованной формулы с применением атома, с которым был связан шаблон этой команды. По существу, транслирующая программа представлена последовательностью нелогических элементов в составе конъюнкта синтеза K_S .

Принципиальная схема решения задачи трансляции на основе транслирующей ПОФ включает четыре этапа. Входными данными являются требования к структуре входных цепочек, входная цепочка как таковая, шаблоны команд запуска транслирующих подпрограмм и дополнительные условия трансляции. По входным данным строится ПОФ. Построенная ПОФ доказывается. По итогам доказательства синтезируется транслирующая программа. Далее транслирующая программа исполняется, а результат ее исполнения записывается в выходную цепочку.

При решении задач трансляции можно строить иерархические композиции транслирующих ПОФ. Транслирующая программа, синтезированная при выводе одной транслирующей ПОФ, может быть использована в качестве одной из транслирующих подпрограмм в транслирующей ПОФ более высокого уровня. В частности, таким образом была решена задача контроля диалога, требования к которому были представлены двухуровневой иерархией формальных языков.

Во **второй главе** диссертации представлены алгоритмы построения распознающих ПОФ для правосторонней регулярной детерминированной грамматики и для LL(1)-грамматики, а также доказаны утверждения об эквивалентности грамматического разбора и процесса доказательства соответствующей ПОФ.

Алгоритм построения распознающей позитивно-образованной формулы $F_R(G, C) = \forall T U_B \{ U_Q$ для регулярной грамматики включает начальную установку позитивно-образованной формулы F_R , циклическую обработку правил грамматики G и циклическую обработку элементов входной цепочки C .

Начальная установка. Если входная строка имеет вид $C = c_1$, положим $U_B = \exists u : u = 1, P(c_1, u), PT(\#), Q(S)$, иначе, если $C = c_1, \dots, c_n, n > 1$, положим $U_B = \exists u : u = 1, P(c_1, u), Q(S)$. Определим множество вопросов одним вопросом $U_Q = \{ \forall : RT(\#) \exists : F$. Здесь атом $P(c_1, u)$ символизирует возможность разбора входного элемента c_1 , расположенного в позиции u входной цепочки; атом $PT(\#)$ означает возможность разбора символа конца входной цепочки $\#$, а атом $Q(S)$ символизирует разбор конструкции, обозначенной нетерминальным символом S (нетерминальным символом S обозначена аксиома грамматики).

Обработка правил грамматики G .

1. Для каждого правила вида $Z \rightarrow aY$ (где $Z \in V_N, Y \in V_N, a \in V_T$) пополним U_Q вопросом $\forall s : P(a, s), Q(Z) \exists : R(a, s), Q(Y)$. Атом $R(a, s)$ означает успешный разбор входного элемента a , расположенного в позиции s входной цепочки.

2. Для каждого правила вида $Z \rightarrow a$ (где $Z \in V_N, a \in V_T$) пополним U_Q вопросом $\forall s : P(a, s), PT(\#), Q(Z) \exists : RT(\#)$. Атом $RT(\#)$ указывает на успешный разбор символа конца входной цепочки.

Обработка элементов входной цепочки $C = c_1, \dots, c_n, n > 1$:

1. Для каждого символа исходной цепочки c_i ($i = 1, \dots, n - 2$) пополним U_Q вопросом $\forall s : s = i, R(c_i, s) \exists : s = i + 1, P(c_{i+1}, s)$;

2. Для символа исходной цепочки c_{n-1} пополним множество U_Q вопросом $\forall s : s = n - 1, R(c_{n-1}, s) \exists : s = n, P(c_n, s), PT(\#)$.

Утверждение 1. Для всякой праволинейной детерминированной регулярной (автоматной) грамматики G и входной цепочки C формула $F_R(G, C)$ доказуема тогда и только тогда, когда $C \in L(G)$, где $L(G)$ означает множество цепочек, порождаемых грамматикой G .

Алгоритм построения ПОФ $F_{LL}(G, C) = \forall T U_B \{ U_Q$ для LL(1)-грамматики также включает в себя начальную установку, циклическую обработку правил грамматики G и циклическую обработку элементов входной цепочки C :

Начальная установка. Определим базу ПОФ $U_B = \exists u : u = 1, P(c_1, u), Q(S), EOP()$. Зададим множество вопросов $U_Q = \{ \forall : PT(\#), EOP() \exists : F$.

Здесь атом $EOP()$ указывает на успешное завершение разбора входной цепочки.

Обработка правил грамматики G .

1. Для каждого правила вида $Z \rightarrow a\gamma$ (где $Z \in V_N$, $a \in V_T$, $\gamma \in V^*$) пополним U_Q вопросом $\forall s: P(a, s), Q(Z) \exists: R(a, s), [\gamma]$. Запись $[\gamma]$ означает преобразование цепочки $\gamma = \gamma_1 \dots \gamma_k \in (V_T \cup V_N)^*$ в конъюнкт по следующим правилам. Если выполнено условие $\gamma_i \in V_T$, то $[\gamma_i] = RT(\gamma_i)$, где атом $RT(\gamma_i)$ символизирует успешный разбор символа γ_i . Если выполнено условие $\gamma_i \in V_N$, то $[\gamma_i] = Q(\gamma_i)$.

2. Обозначим через $DS(Z, \gamma)$ множество направляющих символов³ правила $Z \rightarrow \gamma$. Для каждого правила вида $Z \rightarrow \gamma$ (где $Z \in V_N$, $\gamma = A\beta$, $A \in V_N$, $\beta \in V^*$) и каждого $t \in DS(Z, \gamma)$ пополним U_Q вопросом $\forall s: P(t, s), Q(Z) \exists: P(t, s), Q(A), [\beta]$.

3. Для каждого терминального символа $t \in V_T$ и правила $Z \rightarrow \alpha t \beta$ (где $Z \in V_N^*$, $\alpha \in V^+$, $\beta \in V^*$) пополним U_Q вопросом $\forall s: P(t, s), RT(t) \exists: R(t, s)$.

4. Введем символ конца исходной цепочки $\#$, $\# \notin (V_T \cup V_N)$. Для каждого правила вида $Z \rightarrow \varepsilon$ и элемента $t \in DS(Z, \varepsilon)$ пополним U_Q вопросом $\forall s: P(t, s), Q(Z) \exists: P(t, s)$, если $t \neq \#$, и вопросом $\forall s: PT(\#), Q(Z) \exists: PT(\#)$, если $t = \#$. Символ ε обозначает пустую цепочку.

Обработка элементов входной цепочки C .

1. Для каждого символа входной цепочки c_i , $i < n$, пополним U_Q вопросом $\forall s: s = i, R(c_i, s) \exists: s = i + 1, P(c_{i+1}, s)$.

2. Для символа c_n пополним U_Q вопросом $\forall s: s = n, R(c_n, s) \exists: PT(\#)$.

Утверждение 2. Для всякой LL(1)-грамматики G и входной цепочки C формула $F_{LL}(G, C)$ доказуема тогда и только тогда, когда $C \in L(G)$.

Утверждения 1 и 2 доказаны математической индукцией по количеству шагов грамматического разбора входной цепочки и количеству шагов вывода соответствующей ПОФ.

Третья глава посвящена инструментальному средству конструирования и применения транслирующих ПОФ в составе прикладных программ. Инструментальное средство выполнено в виде программного компонента. Под компонентом понимается объект. Данный объект доступен при проектировании программы (в системе визуального объектно-ориентированного программирования) и при ее исполнении.

³ См., например, Ахо А., Сети Р., Ульман Д. Компиляторы: принципы, технологии и инструменты: Пер. с англ. М.: Изд. дом «Вильямс», 2001. 768 с.

Компонент синтеза транслирующих программ обеспечивает:

- конструирование и испытания на примерах транслирующих ПОФ при проектировании прикладной программы;
- доказательство существования транслирующей программы для заданной входной цепочки языка, синтез и исполнение транслирующей программы при исполнении прикладной программы.

Компонент внедрен в инструментальную среду .Net Framework и используется в прикладной программе на любом из доступных языков платформы .Net.

Основу КСТП составляют объектные представления транслирующей ПОФ и формальной грамматики. Элементами объектного представления транслирующей ПОФ являются экземпляры классов, характеризующих следующие сущности: транслирующая ПОФ в целом; входная и выходная цепочки; транслирующая программа; транслирующие подпрограммы и команды их запуска; атомы, переменные вывода, а также формальные параметры атомов.

Все эти экземпляры классов представлены коллекциями. Коллекции формируются, например, из атомов, составляющих конъюнкт, или из аргументов атома. Транслирующая ПОФ в целом представлена следующими коллекциями:

- переменных вывода при корневом кванторе существования транслирующей ПОФ;
- пар атомов и связанных с ними команд запуска транслирующих подпрограмм в составе конъюнкта при корневом кванторе существования транслирующей ПОФ;
- вопросов в составе формулы транслирующей ПОФ.

Каждый вопрос, в свою очередь, это совокупность коллекций: параметров вопроса; атомов в составе предусловия вопроса; пар атомов и связанных с ними шаблонов команд запуска транслирующих подпрограмм в составе постусловия вопроса.

Формальная грамматика в КСТП представлена классом контекстно-свободных грамматик и его наследниками – классами регулярных и LL(1)-грамматик. Классы регулярных и LL(1)-грамматик отличаются друг от друга функциональными членами, предназначенными для контроля и преобразования грамматики в ПОФ.

Управление экземпляром КСТП при исполнении прикладной программы обеспечивается обращением к его функциональным членам. Среди функциональных членов КСТП основными являются методы Prove, Synthesize и Execute, предназначенные для осуществления доказательства ПОФ, синтеза транслирующей программы и исполнения транслирующей программы соответственно.

Доказательство существования транслирующей программы может осуществляться в непрерывном режиме, когда входная цепочка

сформирована в полном объеме, и в пошаговом режиме – по мере появления элементов входной цепочки.

Синтез и исполнение транслирующей программы также может осуществляться в двух режимах: по мере выполнения успешных шагов доказательства ПОФ или после завершения доказательства.

КСТП разработан в MS Visual Studio.Net 2005 и реализован на языке C#.

Ниже представлен сравнительный анализ КСТП с другими средствами разработки трансляторов.

	КСТП	Prolog	YACC
Решение задач лексического анализа	+	+	-
Решение задачи синтаксического анализа	+	+	+
Работа с LL-грамматиками	+	+	-
Интеграция в прикладную программу	В виде программного компонента	Вызов внешнего интерпретатора Prolog	В виде сгенерированного кода
Размер исходного кода транслятора для задачи распознавания арифметических выражений ⁴	~24 вопроса	~35 предложений	~40 строк
Отладка транслятора	+	-	-
Автоматическое исправление ошибок при трансляции	+	+	-
Графический интерфейс	+	-	-

Четвертая глава посвящена применению КСТП для решения практических задач.

Первой задачей является задача конструирования учебных трансляторов. Конструирование трансляторов является частью учебных курсов по соответствующим дисциплинам. Учебный транслятор представляет собой GUI-приложение платформы .Net. Для решения этой задачи транслятор задается несколькими экземплярами КСТП (по одному для каждого типа лексем и один для синтаксиса входного языка) и программным кодом объемом около 100 строк исходного текста для управления процессом трансляции. Все шаги этого процесса (разбор входных цепочек, синтез и исполнение транслирующих программ) обеспечиваются обращением к функциональным членам соответствующих экземпляров КСТП.

Конструирование транслирующих позитивно-образованных формул обеспечивается соответствующими экземплярами КСТП при проектировании учебного транслятора в инструментальной среде Microsoft Visual Studio.Net.

Все остальные действия, связанные с разработкой учебного транслятора, обеспечиваются штатными средствами интегрированной среды разработки Microsoft Visual Studio.Net.

⁴ Cohen J., Hickey T.J. Parsing and compiling using Prolog // ACM Transactions on Programming Languages and Systems (TOPLAS). 1987. Vol. 9, Issue 2. P. 125–163.

Технология решения учебных задач трансляции в среде Microsoft Visual Studio.Net внедрена в Институте математики, экономики и информатики Иркутского государственного университета. Набор компонентов, обеспечивающих данную технологию, включает КСТП и зарегистрирован Иркутским государственным университетом в Федеральной службе по интеллектуальной собственности, патентам и товарным знакам.

Второй задачей является задача контроля допустимости цепочек входных воздействий в составе диалогового средства таможенного контроля Терминал. Это средство представляет собой мобильное вычислительное устройство, оснащенное программой сбора учетных данных и результатов контрольных измерений объемов для необработанных лесоматериалов.

Для решения данной задачи разработана логико-лингвистическая модель, которая представляет собой комплект из 49 транслирующих позитивно-образованных формул. Транслирующие ПОФ разработаны для каждого элемента управления и диалоговой формы и реализованы в виде экземпляров КСТП. Особенность применения КСТП в данном случае заключается в том, что элементами алфавитов формальных языков, которым принадлежат правильные цепочки входных воздействий, являются corteжy. Corteж состоит из идентификатора элемента диалога, идентификатора события, идентификатора параметра и значения параметра. Доказательство существования, синтез и исполнение транслирующих программ осуществляются в пошаговом режиме по мере формирования входных цепочек.

В соответствии с решением коллегии Федеральной таможенной службы России средство таможенного контроля Терминал принято к экспериментальному опробованию в составе новой технологии контроля необработанных лесоматериалов⁵.

В **заключении** обсуждаются перспективы применения и развития разработанного в диссертации аппарата транслирующих позитивно-образованных формул.

В **приложениях** к диссертации представлены справочная информация по множествам направляющих символов и акт об использовании результатов работы.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

В рамках диссертации получены и выносятся на защиту следующие результаты:

1. Разработаны подмножество языка ПОФ, стратегия логического вывода τ , постановка и принципиальная схема решения задачи трансляции

⁵ О распространении системы электронного поштучного учета: Письмо Федеральной таможенной службы России от 11 марта 2008 г. № 01-11/9055.

на основе исчисления ПОФ. Разработаны алгоритмы построения ПОФ по грамматикам регулярных и LL(1)-языков. Доказана эквивалентность грамматического разбора этих языков процессу вывода ПОФ.

2. Создано инструментальное средство для конструирования и применения транслирующих ПОФ в составе прикладных программ.

3. Разработана постановка и схема решения задачи контроля диалога человека и объектной программы на основе формальной спецификации требований к диалогу в виде двухуровневой иерархии формальных языков, которые задаются ПОФ.

ОСНОВНЫЕ ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Бутаков М.И., Курганский В.И. О решении задачи трансляции планированием вычислений на основе позитивно-образованных формул // Системы управления и информационные технологии. 2007. № 3.1 (29). С. 120–123.

2. Бутаков М.И., Курганская О.В. Решение учебных задач трансляции на основе позитивно-образованных формул // Системы управления и информационные технологии. 2008. № 3.1 (33). С. 124–128.

3. Бутаков М.И., Курганский В.И. Контроль диалога объектных программ на основе позитивно-образованных формул // Вопросы современной науки и практики. Университет им. В.И. Вернадского. 2010. № 4–6 (29). С. 106–115.

4. Курганский В.И., Бутаков М.И. Лексический анализ на основе регулярных грамматик и конечных автоматов // Иркутск: Иркут. гос. ун-т. 2006. 20 с.

5. Курганский В.И., Бутаков М.И. Синтаксический анализ на основе LL(1)-грамматик и автоматов с магазинной памятью. Иркутск: Иркут. гос. ун-т, 2006. 32 с.

6. Бутаков М.И. Компонент построения и исполнения транслирующих программ на основе позитивно-образованных формул // Материалы IX школы-семинара молодых ученых «Математическое моделирование и информационные технологии» (Иркутск, 22–27 октября 2007 г.). Иркутск: РИО ИДСТУ СО РАН, 2007. С. 36–39.

7. Бутаков М.И., Курганский В.И. Об одной модели трансляции на основе позитивно-образованных формул // Информационные технологии моделирования и управления. 2007. № 6 (40). С. 663–672.

8. Бутаков М.И., Курганская О.В. Лексический анализ: обучение моделированию и программированию // Информационные технологии моделирования и управления. 2008. № 2 (45). С. 128–135.

9. Корольков Ю.Д., Курганский В.И., Бутаков М.И. Набор компонентов ParseUnit: Свидетельство об официальной регистрации программы для ЭВМ

№ 2007611658 от 20.04.2007 г. М.: Федеральная служба по интеллектуальной собственности, патентам и товарным знакам, 2007.

10. Бутаков М.И. Компонент построения и исполнения транслирующих программ на основе позитивно-образованных формул // Прикладные информационные технологии и системы. Иркутск: Изд-во Иркут. гос. ун-та, 2009. С. 4–27.

11. Бутаков М.И., Контроль входных воздействий в специализированной системе // Тр. XIV Байкальской Всерос. конф. «Информационные и математические технологии в науке и управлении». Ч. II. Иркутск: ИСЭМ СО РАН, 2009. С. 105–115.

12. Бутаков М.И. Инструментальное средство синтеза и исполнения транслирующих программ на основе позитивно-образованных формул // Материалы 4-й Всерос. мультikonф. по проблемам управления. Таганрог: Изд-во ТТИ ЮФУ, 2011. Т. 1. С. 27–29.

Редакционно-издательский отдел
Учреждения Российской академии наук
Института динамики систем и теории управления
Сибирского отделения РАН
664033, Иркутск, ул. Лермонтова, д. 134
E-mail: rio@icc.ru

Подписано к печати 22.12.2011 г.
Формат бумаги 60×84 1/16, объем 1,2 п.л.
Заказ 36. Тираж 120 экз.

Отпечатано в ИДСТУ СО РАН

