

ПРОЕКТИРОВАНИЕ ПРЕДМЕТНО-ОРИЕНТИРОВАННОГО ЯЗЫКА ДЛЯ РАСЧЕТА СПЕЦИАЛЬНЫХ ФУНКЦИЙ ВОЛНОВЫХ КАТАСТРОФ²

Рассмотрен процесс проектирования предметно-ориентированного языка записи систем обыкновенных дифференциальных уравнений, описывающих СВК с целью их последующего численного решения на ЭВМ. Представлены выдержки из описания формальной грамматики языка, описан процесс интеграции данного языка со специализированной системой численного расчета. Рассмотрены вопросы автоматического распараллеливания решений, записанных на данном языке, их исполнения в распределенной среде.

Ключевые слова: волновые катастрофы, специальные функции, предметно-ориентированный язык, грамматика, обыкновенные дифференциальные уравнения

DESIGNING OF OBJECT-ORIENTED LANGUAGE FOR CALCULATION OF WAVE CATASTROPHES SPECIAL FUNCTIONS

This paper describes the development process of a domain specific language (DSL) for description of ODE systems describing special functions of wave catastrophes for calculation using digital computers. Formal grammar of the language is described in the work on a par with the process of integration of the DSL in the special computation software. Problems of automatic parallel distributed computing in connection with the language are described.

Keywords: wave disaster, special functions, object-oriented language, grammar, ordinary differential equations.

Введение

Расчет специальных функций волновых катастроф, анализ особенностей, описываемых СВК, имеют практическое значение для таких областей, как радиолокация, передача информации средствами волоконно-оптических линий связи, формирование направленных пучков в плазме. Увеличение производительности аппаратных средств современных ЭВМ, а также широкое распространение идей параллельной и распределенной обработки данных предоставляют новые возможности эффективного машинного расчета специальных функций волновых катастроф. В настоящей работе рассматривается проблема создания комплекса программ, производящего вычисление и визуализацию специальных функций широкого ряда известных особенностей при помощи метода систем обыкновенных дифференциальных уравнений (СОДУ) [1; 2]. С целью достижения такого кри-

терия, как простота расширения возможностей программного комплекса, поставлена задача о проектировании предметно-ориентированного языка для описания СОДУ, полученных путем вывода из систем канонических дифференциальных уравнений. Разработка специализированного языка, эффективного окружения и среды исполнения данного языка – важная задача при построении систем подобного класса. Интеграция предметно-ориентированного языка в систему позволит упростить процесс добавления программ расчета новых СВК, снизить порог вхождения разработчиков этих программ за счет использования языка более высокого уровня, отделить описание самого процесса вычисления от прочих действий, связанных с взаимодействием со средой исполнения.

Метод систем обыкновенных дифференциальных уравнений и его реализация в программном комплексе

Волновая теория катастроф предлагает целый ряд методов для изучения структурно-устойчивых фокусировок. К ним можно отнести метод суммирования рядов Тейлора, метод

¹ Аспирант НОУ ВПО «Российский новый университет».

² Работа выполнена при поддержке РФФИ (гранты № 12-02-00413-а, № 13-07-00937, ОФИ_М № 13-02-12121).

равномерных асимптотик, контурный метод. Данные методы достаточно эффективны при расчете СВК катастроф с малым количеством параметров. В случае многопараметрических катастроф [3; 4] сложность применения данных методов значительно возрастает.

При разработке программного комплекса было принято решение использовать метод систем обыкновенных дифференциальных уравнений [1]. Данный метод хорошо применим для расчета СВК как простых, так и для многопараметрических катастроф.

Из системы канонических дифференциальных уравнений в частных производных можно вывести систему обыкновенных дифференциальных уравнений. Переход к такой системе приводит к тому, что помимо вычисления самой СВК требуется вычислить ряд ее первых производных.

Решение системы представляется в виде фундаментального вектора (1). Компоненты вектора обладают следующими свойствами: размерность вектора минимальна из всех возможных, все остальные производные СВК $V_{\Sigma}(S)$ по S_i выражаются с помощью линейных алгебраических соотношений:

$$\vec{W} = (V_{\Sigma}, V_{\Sigma}^1, V_{\Sigma}^2, \dots, V_{\Sigma}^{r-1}). \quad (1)$$

В качестве примера рассмотрим особенность A_3 , СВК которой имеет вид:

$$V_{A_3} = \int_{-\infty}^{+\infty} \exp[i(x^3 + \lambda_2 x^2 + \lambda_2 x)] dx. \quad (2)$$

Фундаментальный вектор будет содержать три компоненты: саму функцию и две ее первые производные.

Система обыкновенных дифференциальных уравнений, соответствующая данной СВК, может быть представлена в следующем виде:

$$\begin{aligned} \frac{dV}{d\lambda_1} &= V^1, & \frac{dV}{d\lambda_2} &= V^2, & \frac{dV^1}{d\lambda_1} &= iV^2, \\ \frac{dV^2}{d\lambda_1} &= \frac{1}{4}(\lambda_1 V - 2i\lambda_2 V^1) \equiv U_{21}, \\ \frac{dV^1}{d\lambda_2} &= U_{21}, & \frac{dV^2}{d\lambda_2} &= -\frac{i}{4}(V + \lambda_1 V^1 + 2\lambda_2 V^2). \end{aligned} \quad (3)$$

Ее дополняют начальные условия:

$$\begin{aligned} V(0) &= \frac{1}{2} \Gamma\left(\frac{1}{4}\right) \exp(i\pi/8), & V^1(0) &= 0, \\ V^2(0) &= \frac{i}{2} \Gamma\left(\frac{3}{4}\right) \exp(i3\pi/8). \end{aligned} \quad (4)$$

Графики амплитуды и фазы данной особенности выглядят следующим образом:

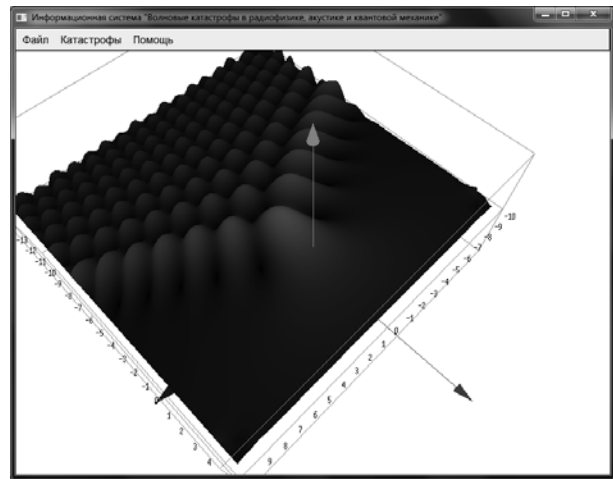


Рис. 1. Особенность A_3 , амплитуда, $\lambda_1 = -15..15, \lambda_2 = -10..1$

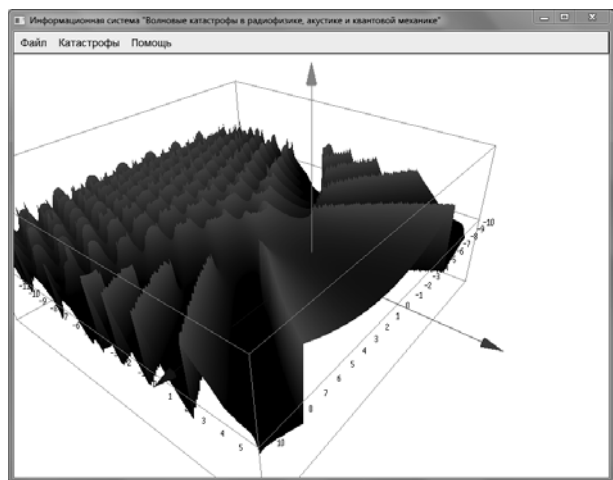


Рис. 2. Особенность A_3 , фаза, $\lambda_1 = -15..15, \lambda_2 = -10..10$

Вычисление СВК сводится к решению системы обыкновенных дифференциальных уравнений известным численным методом (см. [5; 6]).

Системы обыкновенных дифференциальных уравнений, описывающих ряд СВК, были получены вручную [1; 6], в то же время, процесс их вывода может быть автоматизирован с применением систем символьных вычислений.

В рамках разрабатываемого программного комплекса вычисления производятся оптимизированной реализацией метода Кутты – Мерсона (4 порядок точности, автоматическое изменение шага [7]). В библиотеке также представлены методы с меньшей точностью, но более высоким быстродействием (метод Рунге – Кутты, метод Эйлера).

Благодаря специальной иерархии классов СВК может быть описана путем наследования родительского класса “Problem”, который уже

реализует общую функциональность, характерную для большинства катастроф. Единственный метод, который требуется определить, – метод “calculate”, выполняющий вычисления. Также определяется ряд функций, описывающих непосредственно системы дифференциальных уравнений.

Вычисление СВК выполняется параллельно за счет разделения задачи на подзадачи, покрывающие свое множество значений параметров. Также возможно вычисление СВК в простой распределенной среде. На данный момент опробована реализация с сетью из двух машин. Ведется работа над организацией сети с множеством вычислительных узлов, множеством потребителей и одним сервером, распределяющим нагрузку (рис. 3).

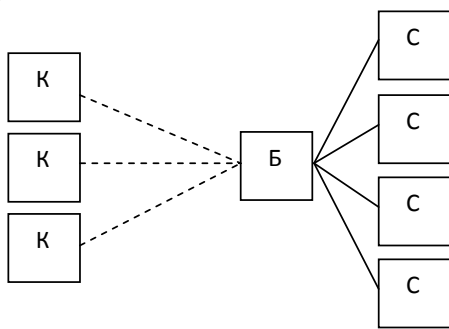


Рис. 3. Схема сети с тремя клиентами, балансировщиком нагрузки и четырьмя серверами

Грамматика предметно-ориентированного языка

С целью улучшения читаемости кода на рассматриваемом языке его грамматика основывается на подходах линии языков Никлауса Вирта [8], поскольку языки с подобной грамматической системой широко распространены в научной среде (Pascal, Modula, Oberon, в некоторой мере Ada, Algol). Тем не менее, грамматики названных языков слишком обширны и мало подходят в качестве модели встраиваемого специализированного языка программирования. С целью упрощения языка в грамматику явно не были включены такие конструкции, как циклы, ветвления, определения записей (структур), в языке отсутствует какая-либо развитая система типов. Программа на целевом языке имеет линейную структуру, не требует вызова каких-либо дополнительных процедур, что позволяет сосредоточиться на решении задачи, абстрактно описывая ее в терминах языка [9].

Описание грамматики языка включает примерно два десятка нетерминальных символов. Ключевые слова представлены как в кириллической, так и в латинской формах. Для ясности

дальнейшего изложения разбирается лишь случай использования кириллических ключевых слов.

Программа, описывающая новую катастрофу, представляет собой модуль, состоящий из секций, каждая из которых есть отражение той или иной системы обыкновенных дифференциальных уравнений, записанных в виде, пригодном для численного решения.

Модуль начинается с ключевого слова «ОСОБЕННОСТЬ», за которым следует строка, описывающая данную катастрофу (обычно – буквенно-числовое обозначение особенности в соответствии с классификацией). Следующий семантический блок модуля – глобальные определения переменных и параметров: каждая переменная и параметр представлены в системе в виде чисел в формате, описываемом стандартом IEEE 754 [10]. Таким образом, нет необходимости вводить какие-либо конструкции, описывающие типы переменных. Список переменных начинается с ключевого слова «ПЕРЕМЕННЫЕ», список параметров – с ключевого слова «ПАРАМЕТРЫ». Важно заметить, что нетерминальные символы, представляющие объявления переменных и параметров для глобальной и локальной областей видимости, – различны. Данный факт связан с тем, что переменные могут иметь различную область видимости, в то время как параметры СВК едины для всей задачи. Ключевое слово «ВЕКТОРЫ» позволяет объявить ряд векторов, используемых в процессе решения. Данное ключевое слово может быть использовано как в глобальной области видимости, так и в локальных областях, образуемых описаниями отдельных систем.

Каждая секция, описывающая СОДУ, задается ключевым словом «СИСТЕМА», за которым следуют буквенно-числовой идентификатор и число уравнений в системе, заданное в круглых скобках. Как уже было замечено, внутри блока «СИСТЕМА» также возможны объявления переменных, тело блока помещается между ключевыми словами «НАЧАЛО» и «КОНЕЦ». Результирующий вектор, получаемый в результате одной итерации (решения системы для одного сочетания параметров), должен быть занесен в специальный вектор, определенный системой исполнения, имя которого «результат».

Основное тело программы следует за определениями систем уравнений, начинается ключевым словом «НАЧАЛО» и завершается ключевым словом «КОНЕЦ», за которым должна следовать точка. В основном теле программы позволяет инициализировать переменные и

векторы, объявленные в глобальной области видимости, а также вызывать подпрограммы библиотеки среды исполнения.

Грамматика языка записывается в расширенной форме Бэкуса – Наура [11] следующим образом:

<p>Особенность = ОСОБЕННОСТЬ строка; ГлобПослОбъявл { СИСТЕМА <i>идент</i> `(` число `)` ЛокПослОбъявл НАЧАЛО ПослОператоров КОНЕЦ } НАЧАЛО ПослОператоров КОНЕЦ `.`</p>
<p>ГлобПослОбъявл = ПАРАМЕТРЫ { СписокОпрПарам `;` } [ПЕРЕМЕННЫЕ { СписокИдент `;` }] [ВЕКТОРЫ { СписокОпрВект `;` }] .</p>
<p>ЛокПослОбъявл = [ПЕРЕМЕННЫЕ { СписокОпрПерем `;` }] [ВЕКТОРЫ { СписокОпрВект `;` }] .</p>
<p>СписокОпрПерем = ОпрПерем { `, ` ОпрПерем } .</p>
<p>ОпрПерем = <i>идент</i> [`<`-` ПрефЧисло ИмпортЗнач] .</p>
<p>СписокОпрВект = ОпрВект { `, ` ОпрВект } .</p>
<p>ОпрВект = <i>идент</i> [число] .</p>
<p>СписокОпрПарам = ОпрПарам { `, ` ОпрПарам } .</p>
<p>ОпрПарам = <i>идент</i> `(` ПрефЧисло ИмпортЗнач, ПрефЧисло ИмпортЗнач, ПрефЧисло ИмпортЗнач `)` <i>идент</i> `(` ПрефЧисло ИмпортЗнач `)` .</p>
<p>ПрефЧисло = [`+` ``-``] <i>число</i> .</p>
<p>ИмпортЗнач = ``\$`` <i>число</i> .</p>
<p>ПослОператоров = Оператор { Оператор } .</p>
<p>Оператор = Обозначение `<`-` Выражение `;` .</p>

Слагаемое = Множитель {ОперУмнож Множитель}.
Множитель = Обозначение [<code>`(` СписВыр `)`</code>] <i>число</i> <code>`(` Выражение `)`</code> .
СписВыр = Выражение {, Выражение}.
Обозначение = <i>идент</i> [<code>`[` Выражение `]`</code>].
Выражение = [<code>`+`</code> <code>`-`</code>] Слагаемое {ОперСлож Слагаемое}.
ОперСлож = <code>`+`</code> <code>`-`</code> .
ОперУмнож = <code>`*`</code> <code>`/`</code> .
<i>идент</i> = буква {буква цифра}.
<i>число</i> = цифра {цифра} <code>`.`</code> {цифра}.
<i>строка</i> = <code>``</code> {СИМВОЛ} <code>``</code> .

Грамматика относится к классу LL(1). Текст программы, записанной с применением данной грамматики, может быть проанализирован с применением метода рекурсивного спуска или другим нисходящим методом. Семантический анализ и генерация кода могут быть выполнены за один проход, фактически исключая такую стадию трансляции, как построение дерева про-

граммы, поскольку в случае метода рекурсивного спуска данная структура данных образуется косвенным образом на стеке вызовов синтаксического анализатора.

Программа, вычисляющая СВК особенности A_3 , записывается в терминах данной грамматики следующим образом:

ОСОБЕННОСТЬ 'A3'

ПАРАМЕТРЫ

lambda1 (\$1, \$2, \$3), lambda2 (\$4, \$5, \$6);

ПЕРЕМЕННЫЕ

gamma1 <- 3.625609908, gamma2 <- 1.225416702;

ВЕКТОРЫ

y[6];

СИСТЕМА A3 (6)

ПЕРЕМЕННЫЕ l1, l2, i21m, i21r, i22m, i22r;

НАЧАЛО

```
l1 <- lambda1 * t;  
l2 <- lambda2 * t;  
i21r <- 0.25 * (l1 * y[1] + 2.0 * l2 * y[4]);  
i21m <- 0.25 * (l1 * y[2] - 2.0 * l2 * y[3]);  
i22r <- 0.25 * (y[2] + l1 * y[4] + 2.0 * l2 * y[6]);  
i22m <- 0.25 * (-y[1] - l1 * y[3] - 2.0 * l2 * y[5]);
```

```
результат[1] <- lambda1 * y[3] + lambda2 * y[5]  
результат[2] <- lambda1 * y[4] + lambda2 * y[6];  
результат[3] <- -lambda1 * y[6] + lambda2 * i21r;  
результат[4] <- lambda1 * y[5] + lambda2 * i21m;  
результат[5] <- lambda1 * i21r + lambda2 * i22r;  
результат[6] <- lambda1 * i21m + lambda2 * i22m;
```

КОНЕЦ

НАЧАЛО

```
y[1] <- 0.5 * gamma1 * cos(PI / 8);  
y[2] <- 0.5 * gamma1 * sin(PI / 8);  
y[3] <- 0;  
y[4] <- 0;  
y[5] <- -0.5 * gamma2 * sin(3 * PI / 8);  
y[6] <- 0.5 * gamma2 * cos(3 * PI / 8);
```

результат <- МетодКуттыМерсона(A3, y);

КОНЕЦ.

Конструкции, начинающиеся со знака “\$” в описании параметров СВК служат для связи программы со средой исполнения, а именно – для импорта значений, предоставляемых средой, например через графический интерфейс.

Процесс вычислений может быть распараллелен автоматически, поскольку в рамках описанной грамматики объявление параметров предшествует основному тексту программы, информация об их количестве и диапазонах значений позволяет заранее разделить задачу на подзадачи.

Среда исполнения

Код описанного предметно-ориентированного языка транслируется в переносимое промежуточное представление, являющееся кодом простой стековой виртуальной машины с фиксированной длиной команды.

Среда исполнения представляет собой совокупность виртуальной машины, способной ис-

полнять код промежуточного представления, и библиотеки окружения, включающей функции численного решения ОДУ, процедуры работы с потоками, примитивами синхронизации и сетью. Код библиотеки окружения написан с использованием языка программирования C++, что гарантирует его быстрое исполнение.

Среда исполнения автоматически создает необходимое число вычислительных контекстов, запускаемых в разных потоках на локальной или удаленных машинах. Таким образом, вычисления выполняются с привлечением всех доступных аппаратных ресурсов.

В процессе разработки виртуальной машины в дальнейшем планируется уделить внимание такому подходу, как JIT-компиляция. Поскольку входной язык не позволяет явно создавать конструкций ветвления или циклов, базовые блоки данного языка достаточно велики, что позволяет организовать эффективную компиляцию времени исполнения. В таком случае, сгенери-

рованный код не должен уступать в производительности коду, разработанному на каком-либо компилируемом языке программирования (например, С).

Выводы

Рассмотрен вопрос проектирования специализированной системы расчета СВК и предметно-ориентированного языка, применяемого в рамках системы для описания катастроф, подпрограммы вычисления которых не заложены в комплекс по умолчанию. Подход с применением специального предметно-ориентированного языка позволяет улучшить переносимость программного пакета, упростить его расширение, увеличить уровень абстракции и скрыть механизмы, связанные с параллелизмом и распределенной обработкой.

На данном этапе разработаны лексический и синтаксический анализаторы входного языка. Ведется работа над генератором кода и виртуальной машиной для последующего внедрения в программный комплекс.

В дальнейшем полученные в работе результаты планируется применить для развития открытой информационной системы «Волновые катастрофы в радиофизике, акустике и квантовой механике» [12–14].

Литература

1. Крюковский А.С. Метод обыкновенных дифференциальных уравнений для расчета специальных функций волновых катастроф (СВК) // Дифракция и распространение электромагнитных волн : межведомственный сборник научных трудов. – М. : Московский физико-технический институт, 1992. – С. 29–48.
2. Крюковский А.С. Равномерная асимптотическая теория краевых и угловых волновых катастроф. – М. : РосНОУ, 2013. – 368 с.
3. Ипатов Е.Б., Крюковский А.С., Лукин Д.С., Палкин Е.А. Краевые катастрофы и асимптотики // ДАН СССР. – 1986. – Т. 291. – № 4. – С. 823–827.
4. Крюковский А.С., Лукин Д.С., Палкин Е.А. Равномерные асимптотики и угловые катастрофы // Доклады РАН. – 1995. – Т. 341. – № 4. – С. 456–459.
5. Дорохина Т.В., Крюковский А.С., Малышенко А.Б. Разработка численных алгоритмов расчета и визуализации волновых катастроф // Вестник РосНОУ. Управление вычислительная техника и информатика. – М. : Российский новый университет, 2008. – Вып. 3. – С. 25–47.
6. Крюковский А.С., Лукин Д.С. Теория расчета эталонных фокальных и дифракционных электромагнитных полей на основе специальных функций волновых катастроф // Радиотехника и электроника. – 2003. – Т. 48. – № 8. – С. 912–921.
7. Shampine L.F. Error Estimation and Control for ODEs // Mathematics Department. Southern Methodist University. – Dallas, 2004.
8. Niklaus Wirth. Module 2 and Oberon // ETH, Zürich, June 2005.
9. Крюковский А.С., Рогачев С.В. Система расчета и визуализации специальных функций волновых катастроф // Электромагнитные волны и электронные системы. – 2013. – Т. 18. – № 8. – С. 10–17.
10. IEEE Standard for Floating-Point Arithmetic, IEEE Std 754-2008, vol., no., pp.1,58, Aug. 29, 2008.
11. Aho, Sethi, Ullman, Compilers: Principles, Techniques, and Tools // Addison-Wesley, 2008. ISBN 978-5-8459-1349-4.
12. Дорохина Т.В., Крюковский А.С., Лукин Д.С. Информационная система «Волновые катастрофы в радиофизике, акустике и квантовой механике» // Электромагнитные волны и электронные системы. – 2007. – Т. 12. – № 8. – С. 71–75.
13. Дорохина Т.В., Крюковский А.С., Лукин Д.С., Волкова Е.В., Костьо А.О., Павлова М.В. Создание информационной системы волновой теории катастроф и ее применение при математическом моделировании // Вестник Российского нового университета, 2007. Выпуск 2. – М. : РосНОУ, 2007. – С. 91–107.
14. Дорохина Т.В., Ипатов Е.Б., Крюковский А.С., Лукин Д.С., Палкин Е.А., Растягачев Д.В. Математическое компьютерное моделирование волновых полей типа катастроф // Распространение радиоволн : сборник докладов XXI Всероссийской научной конференции, Йошкар-Ола, 25–27 мая 2005 г. – Йошкар-Ола : МарГТУ, 2005. – Т. 2. – С. 336–339.