САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ

На правах рукописи

ТЕРЕНТЬЕВ Сергей Валерьевич

РАЗРАБОТКА И РЕАЛИЗАЦИЯ ОСНОВАННЫХ НА ИНТЕРВАЛЬНОЙ АРИФМЕТИКЕ АЛГОРИТМОВ КОМПЬЮТЕРНОГО ИССЛЕДОВАНИЯ ДИНАМИЧЕСКИХ СИСТЕМ

05.13.11 — Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата физико-математических наук

Санкт-Петербург
2010

Работа выполнена на кафедре информатики Математико-механического факультета Санкт-Петербургского государственного университета.

Научный руководитель:	кандидат физико-математических наук, доцент АМПИЛОВА Наталья Борисовна
Официальные оппоненты:	доктор физико-математических наук, профессор ФЛЕГОНТОВ Александр Владимирович (Российский государственный педагогический университет им. А.И. Герцена),
	доктор физико-математических наук, профессор АНДРИАНОВ Сергей Николаевич (Санкт-Петербургский государственный университет).
Ведущая организация:	Санкт-Петербургский государственный политехнический университет.

Защита состоится "____" ________ 2010 года в ____ часов на заседании совета Д 212.232.51 по защите докторских и кандидатских диссертаций при Санкт-Петербургском государственном университете по адресу 198504, Санкт-Петербург, Петродворец, Университетский пр.,28, Математико-механический факультет.

С диссертацией можно ознакомиться в Научной библиотеке им.М.Горького Санкт-Петербургского государственного университета по адресу: 199034, Санкт-Петербург, Университетская наб. 7/9.

Автореферат разослан "____" ________ 2010 г.

Ученый секретарь
диссертационного совета Даугавет И. К.

Общая характеристика работы

Актуальность темы. Изучение поведения сложных динамических систем как основных моделей многих реальных процессов требует создания современных компьютерно-ориентированных алгоритмов исследования. Наиболее важной характеристикой поведения систем является существование у них инвариантных множеств.

В теории динамических систем в центре внимания стоит изучение асимптотического поведения, т.е. поведения системы при устремлении времени к бесконечности, особенно при наличии нетривиального возвращения. Ставится ряд вопросов об эволюции системы, например, может ли система прийти в одно из состояний, в которых она была раньше; насколько похожи процессы эволюции системы при близких начальных состояниях.

Развитие компьютерной техники привело к активному использованию компьютерного моделирования при изучении динамики систем со сложным поведением траекторий. Один из основных классов компьютерных методов исследования динамических систем представляют так называемые методы, основанные на множествах (set-oriented methods), которые используют конечное покрытие фазового пространства набором ячеек (клеток) и отслеживают динамику системы по попаданию точек траекторий исследуемой системы в эти клетки. Для выбранной области фазового пространства K, строится последовательное подразбиение ячеек, удаляются те ячейки, образы которых заведомо не принадлежат K. При стремлении диаметра ячеек к нулю мы можем получать все более точный фазовый портрет. Поскольку образы ячеек могут вычисляться независимо, реализация таких методов может использовать параллельные вычисления [11]. На этих методах основано большинство алгоритмов локализации разных видов инвариантных множеств и, в частности, аттракторов динамических систем ([16, 17, 18, 19]).

Построение образов ячеек при реализации описанных методов приводит к необходимости обозначать факт пересечения образа с остальными ячейками покрытия. Весьма удачным оказался метод символического образа, предложенный в 1983 году Г.С. Осипенко [4]. Символический образ есть конечная аппроксимация динамической системы, представляющая собой ориентированный граф [3, 10]. Он строится по заданному покрытию фазового пространства ячейками C_i, вершины графа соответствуют ячейкам, дуги — связям между ними, а именно: вершины i и j соединяются дугой, если образ ячейки C_i при действии динамической системы пересекается с ячейкой C_j. В работах [1, 4, 5, 6, 8, 22, 23] приводятся доказательства сходимости метода при решении различных задач, например, построении инвариантных, в частности, цепно-рекуррентных множеств динамических систем.

Полученная при последовательном подразбиении ячеек покрытия последовательность символических образов позволяет получить приближение к динамике системы, при этом точность построения оценивается через параметры символического образа. Применение метода символического образа к исследованию динамических систем описано в работах [1, 4, 5, 8, 15, 23, 24].

Существенным недостатком алгоритмов локализации инвариантных множеств, основанных на обычной арифметике является их экспоненциальная сложность, что делает их применение затруднительным для систем размерности большей двух.

Наряду с развитием методов, основанных на обычной арифметике, все чаще появляются работы, использующие арифметику интервальную, основанную на проведении вычислений с получением точных границ для искомого значения, т.е. получения в качестве ответа интервала, содержащего это значение. К настоящему моменту описано применение интервальной к исследованию динамических систем при нахождение периодических орбит, локализации неблуждающих множеств и выявление у системы символической динамики. Актуальной является задача создания программного комплекса для исследования динамических систем, основанного на интервальной арифметике и позволяющего решать задачи локализации инвариантных множеств.

Данная работа посвящена разработке и реализации алгоритмов исследования динамических систем, основанных на использовании методов интервальной арифметики и символического образа. Ячейка покрытия естественным образом может быть представлена интервальным вектором, а интервальное расширение функций, описывающих систему, можно рассматривать как отображение символического образа. При этом граф не строится, существование дуги отмечается в специальной структуре данных. Разработаны и реализованы два способа оптимизации представления данных. Реализовано несколько интервальных расширений вещественной функции. Для использования динамической генерации кода был разработан язык описания динамической системы, реализован его компилятор. Проведены численные эксперименты, которые показывают преимущества интервального метода, скорость выполнения возрастает в несколько раз.

Основные задачи.

Локализация инвариантных множеств. В работе разработаны и реализованы версии алгоритма с перечисленными ниже оптимизациями.

1. **Представление исходных данных**. Используется целочисленная система координат, все ячейки имеют одинаковый размер, каждая представлена вектором координат левого верхнего угла. Реализовано 2 способа хранения информации о ячейке: а) сам координатный вектор, и б)номер ячейки, полученный с помощью преобразования многомерного индекса в одномерный. При первом способе хранения для оптимизации алгоритма локализации инвариантных множеств используется специальная структура хранения данных — R–деревья. Во втором случае каждая ячейка представлена одним битом, означающим ее присутствие в получаемом приближении к инвариантному множеству. (В начале работы все ячейки получают признак 1.) При таком способе хранения существенно уменьшается объем используемой оперативной памяти.

2. **Вычисление арифметических выражений**. В отличие от традиционных методов не строится дерево разбора, а используется специально разработанный язык описания динамических систем и также компилятор [2], позволяющий на основе описания исходной системы динамически получить библиотеку, которая содержит методы доступа к описанию и вычислению арифметических выражений. Библиотека может быть динамически загружена в адресное пространство пользовательского процесса для дальнейшего использования. Разработанный компилятор является кросс-платформенным. Реализована поддержка как обычной и так и интервальной арифметики.

3. **Поиск ячейки в покрытии**. Было разработано и реализовано 2 способа индексации ячеек покрытия: многомерный с использованием R-деревьев и одномерный. В первом способе, при поиске ячейки, производится динамическое отсечение областей покрытия, в которых ячейки заведомо нет. В случае одномерной индексации, поиск ячейки сводится к обращению к элементу массива по адресу, однозначно определяемому по индексу ячейки.

Также в программном комплексе был реализован распределенный алгоритм локализации инвариантных множеств, позволивший увеличить эффективность вычислений.

Построение псевдотраекторий, проходящих через 2 заданные точки. Интервальные методы позволяют достаточно просто решать задачу о существовании приближенной траектории, проходящей через две заданные точки. Кроме очевидного решения (отрезка точной траектории или траектории с началом в первой точке и проходящей вблизи второй), этот метод позволяет получить в качестве ответа последовательность интервальных векторов, таких, что любая траектория, составленная из точек, выбранных по одной из каждого такого множества, является ε-траекторией с некоторым ε (заданным или определенным в процессе решения).

С помощью малого возмущения исходной системы можно также получить приближенные траектории возмущенной системы и оценить, при каких возмущениях они будут являться точными траекториями исходной. Такого рода вопросы часто возникают в задачах теории управления.

Для эффективной работы с арифметическими выражениями в реализацию данного алгоритма была использована динамическая генерация кода. При вычислении константы Липшица для вычисления нормы интервала используется распределенный алгоритм.

Цель работы. Целью работы является разработка и реализация алгоритмов компьютерного исследования динамических систем, основанных на использовании интервального анализа и методов символического образа, которые позволят решать следующие классы задач:

- локализация инвариантных множеств дискретных и непрерывных динамических систем;
- построение множества псевдотраекторий динамической системы, проходящих через две заданные точки.

Основные результаты. В работе получены следующие основные научные результаты:

1. Разработаны и реализованы алгоритмы компьютерного исследования динамических систем на основе методов интервальной арифметики и символического образа. С их помощью решаются задача локализации инвариантных множеств дискретных и непрерывных динамических систем, в которой разработана и реализована специальная структура хранения дуг графа и задача построения множества псевдотраекторий динамической системы, проходящих через две

заданные точки (для дискретных и непрерывных динамических систем). Для повышения быстродействия обработки входных параметров системы разработан язык описания динамических систем, а также реализован его компилятор. Такой подход позволяет избежать традиционного обхода дерева арифметического выражения при вычислении значении правых частей. Реализованный программный продукт может быть расширен новыми алгоритмами исследования динамических систем. В свою очередь компоненты данного программного комплекса могут быть использованы как части другой системы.

2. Разработан и реализован интервальный алгоритм локализации инвариантных множеств. Доказано, что его временная сложность есть $O(n*T)$, где n — число ячеек в текущем покрытии, T — время поиска ячейки в покрытии. Если информация о ячейке представлена номером, полученным с помощью преобразования многомерного индекса в одномерный, тогда временная сложность алгоритма есть $O(n)$. Тем самым показано преимущество метода в сравнении с методами, основанными на обычной арифметике, так как они имеют экспоненциальную сложность. Реализовано несколько версий алгоритма со следующими оптимизациями:

   - представление исходных данных;
   - вычисление арифметических выражений;
   - операция поиска ячейки.

3. При решении задачи о существовании псевдотраекторий, проходящих через две заданные точки, разработаны и реализованы два метода построения ε-траекторий. Доказаны теоремы об их сложности. Для эффективной работы была внедрена динамическая генерация кода. При вычислении константы Липшица для вычисления нормы интервала разработан и реализован распределенный алгоритм.

Научная новизна. Все основные научные результаты диссертации являются новыми. Разработаны и реализованы алгоритмы компьютерного исследования динамических систем на основе интервальной арифметики и метода символического образа, которые объединяют в себе: алгоритм локализации инвариантных множеств дискретных и непрерывных динамических систем, в котором разработана и реализована специальная структура хранения дуг графа; метод построения псевдотраекторий проходящих через две заданные точки (для дискретных и непрерывных динамических систем). Разработан язык описания динамических систем, а также реализован его компилятор. Показано , что грамматика этого языка является LL-грамматикой, т.е. левосторонний анализ можно представить в виде работы детерминированного конечного автомата, достаточно простого в реализации.

Теоретическая ценность и практическая значимость.

Разработан и проанализирован интервальный алгоритм локализации инвариантных множеств. Доказано, что его сложность равна $O(n * T)$, где n — число ячеек в текущем покрытии, T — время поиска ячейки в покрытии. Разработаны и реализованы 2 способа хранения данных. Показано, что во всех случая время выполнения сокращается в несколько раз по сравнению с алгоритмами, использующими обычную арифметику.

Разработанные алгоритмы компьютерного исследования динамических систем на основе метода символического образа и интервальной арифметики могут быть применены для исследования непрерывных и дискретных динамических систем любой размерности. Они используется в курсе по компьютерному исследованию динамических систем на математическом факультете СПбГУ, а также могут быть использованы для исследования нетривиальных динамических систем и в учебном процессе.

Апробация работы. Результаты работы докладывались:

- на конкурсе–конференции студентов, аспирантов и молодых ученых Северо–Запада“Технологии Microsoft в теории и практике программирования“(Санкт-Петербург, 2007);

- на научной международной конференции „Космос, астрономия и программирование (Лавровские чтения)“(Санкт-Петербург, 2008);

- на научно–технической конференции „Научное программное обечпечение в образовании и научных исследованиях“(Санкт-Петербург, 2008);

- на третьей Международной научно–практической конференции „Современные информационные технологии и IT-образование“(Москва, 2008);

- на 5-й международной конференции „Dynamical Systems and Applications“ (Constantza, Romania, 2009);

- на XLI международной научная конференция аспирантов и студентов „Процессы управления и устойчивость“ Control Processes and Stability (CPS'10) (Санкт-Петербург, 2010)

Публикации. Основные результаты диссертации опубликованы в работах [1, 2, 3, 4, 6, 8, 9]. Из них две публикации [1, 2] в журналах из перечня ВАК. Работы [1, 3, 4] написаны в соавторстве. В работах [1, 3, 4] Н.Б.Ампиловой принадлежат общие постановки задач, а С.В.Терентьеву – реализация описываемых методов, создание демонстрационных примеров и программных средств. В [3] Е.И. Петренко также является автором некоторых описываемых методов.

Структура и объем диссертации. Диссертация состоит из введения, четырех глав, заключения, списка литературы, включающего 76 источников, двух приложений. Текст занимает 143 страницы, содержит 32 рисунка и три таблицы.

Содержание работы

Введение содержит обзор современного состояния данной предметной области, обоснование актуальности диссертационной работы.

Во введении сформулированы цели и аргументирована научная новизна исследований, представлены выносимые на защиту положения.

Первые две главы работы содержат описание решенных при разработке алгоритмов компьютерного исследования динамических систем задач и результаты численных экспериментов.

Третья глава посвящена специально разработанному языку описания динамических систем и схеме работы модуля динамической генерации кода.

В четвертой главе приведено описание особенностей реализации созданного программного комплекса.

Первая глава содержит основные понятия интервальной арифметики, используемые в данной работе, описание метода символического образа, а также разработанное автором представление графа символического образа, используемое в задаче локализации. Также здесь описан алгоритм локализации инвариантных множеств, основанный на использовании интервальной арифметики.

Символический образ представляет собой ориентированный граф, который строится по дискретной динамической системе $x_{n+1} = f(x_n)$, $f : D \subset R^m \to R^m$, f — диффеоморфизм, или по отображению сдвига для непрерывной динамической системы ($\dot{x} = g(x)$), и разбиению множества D на конечный набор ячеек $\{C_i\}$. Вершинам графа соответствуют ячейки, между вершинами i и j существует дуга $i \to j$, тогда и только тогда, когда $f(C_i) \cap C_j \neq \emptyset$.

В реализации алгоритма граф не строится, рассматривается его множество-представление. Пусть $D_1 = \left\{\breve{M}_1, \ldots, \breve{M}_n\right\}$ — начальное покрытие компакта $\breve{M}$ ячейками $\breve{M}_i \subseteq IR^m; i = 1, \ldots, n$. Для каждой ячейки $\breve{M}_i$ вычислим множества $\breve{R}_i = \bigcup_{j \in c_i} \breve{M}_j$, где c_i — множество индексов j тех ячеек, для которых $f(\breve{M}_i) \bigcap f(\breve{M}_j) = \emptyset$. Построим множество-представление графа G_1, соответствующее символическому образу G_1 отображения f относительно покрытия D_1, а именно $G_{D_1} = \bigcup_{i \in [1,n]} \breve{R}_i$. Это множество содержит все ячейки покрытия, которые участвуют в построении графа G_1. На следующем шаге разбиение применяется к элементам множества G_{D_1}, в результате повторения описанной процедуры получается множество-представление G_{D_2} для графа G_2.

На k-м шаге рассматриваем разбиение вида $D_k = \left\{\breve{M}_{j_1}, \ldots, \breve{M}_{j_m} \mid \breve{M}_{j_l} \subset G_{D_{k-1}}\right\}$, строится множество-представление G_{D_k}. Погрешность аппроксимации инвариантного множества может быть оценена через параметры символического образа. Ниже представлено краткое описание алгоритма построения множества G_{D_k}:

1. Пусть $G_{D_{k-1}}$ — текущее множество-представление графа G,
   а множество G_{D_k} – уточнее $G_{D_{k-1}}$.

2. Пусть $G_{D_k} = \emptyset$.

3. Для каждой ячейки из $G_{D_{k-1}}$

   - вычислить ее образ: Img;
   - вычислить пересечение Img с $G_{D_{k-1}}$: Int_{Img};
   - Если $Int_{Img} \neq \emptyset$ тогда $G_{D_k} = G_{D_k} \bigcup Int_{Img}$.

4. Вернуть в качестве результата множество G_{D_k}.

В реализации алгоритма рассматриваются одинаковые ячейки, каждая ячейка представляется координатой ее верхнего левого угла. Множество D представляет собой параллелепипед, ориентированный по осям координат. Координатные оси пространства R^m разбиваются на части одинаковой длины, так, чтобы по i-ому направлению множество D разбивалось на p_i частей. Рассматривается система координат, за единицу длины в которой принимается размер ячейки. Каждой ячейке сопоставляется набор из m целых чисел.

Доказано следующее

Утверждение.

Время работы алгоритма локализации инвариантных множеств равно $O(n{*}T)$, где n — число ячеек в покрытии D_k, T — время поиска ячейки в покрытии.

При сравнении интервального алгоритма локализации с методами локализации инвариантных множеств, основанных на обычной арифметике, было установлено, что скорость локализации данного подхода существенно выше.

Во второй главе описано применение интервальных методов для решения задачи о существовании ε-траектории дискретной или непрерывной динамической системы (с заданным или вычисленным ε), проходящей через 2 заданные точки. В случае, если такая траектория существует, можно также получить последовательность интервалов (A), так что любая последовательность точек, выбираемых произвольно в каждом из них, является ε-траекторией. Для получения такой последовательности были разработаны и реализованы два алгоритма: алгоритм построения δ-траекторий с $\delta = \varepsilon$ и с $\delta \in [\varepsilon, \varepsilon + q]$, где q — максимальный из диаметров образов ячеек.

Доказано следующее

Утверждение.

Время работы алгоритмов построения множества приближенных траекторий равно $O(N)$, где $N = |A|$. С помощью описанного метода можно численно определить параметры, при которых приближенная траектория возмущенной системы является точной траекторией исходной.

Приведены результаты численных экспериментов.

В третьей главе описан способ реализации смешанных вычислений [14]. Этот термин был введен для обозначения техники оптимизации компьютерных программ. Мы используем эту технику при реализации модуля динамической генерации кода, так как при реализации алгоритмов исследования динамических систем возникает необходимость в эффективной многократной однообразной обработке параметров и функции, описывающих конкретную систему. Для решения этой задачи был разработан язык $DYNAMIC\ SYSTEM\ DEFINITION\ LANGUAGE$ (

$DSDL$), а также реализован компилятор для него.

Доказана следующая

Теорема.

Порождающая грамматика (G_{dsdl}) *языка* $DSDL$ *является контекстно-свободной LL-грамматикой.*

Это значит, что существует класс LL-анализаторов, способных разобрать входную цепочку, порожденную с помощью G_{dsdl}. На практике большим преимуществом LL-разбора является простота реализации, так как левосторонний анализ можно представить в виде работы детерминированного конечного автомата. Подробно описана схема работы модуля динамической генерации кода, а также обсуждаются особенности процесса трансляции описания динамических систем.

В четвертой главе главе приведены технические особенности реализации разработанных алгоритмов для исследования динамических систем. В работе реализованы 2 библиотеки динамической генерации кода:

- для Microsoft .Net платформы;
- с поддержкой POSIX стандартов.

Так как реализация для POSIX стандартов является наиболее трудоъемкой (для .Net платформы существует встроенная высокоуровневая модель компилятора и компоновщика — .Net Code Document Object Model), в этой главе подробно обсуждаются особенности организации процесса компиляции в промежуточный код с использованием набора инструментов $MinGW$[21](приведены примеры использования библиотек).

Алгоритм локализации инвариантных множеств реализован на языке C++. При реализации алгоритма была использована библиотека BOOST интервальной арифметики [12]. Визуализация инвариантных множеств осуществляется с помощью технологии GNUPLOT [7]. Распределенный алгоритм локализации инвариантных множеств ДС реализован с помощью библиотеки Boost.Threads, которая входит в состав Boost — собрание библиотек, расширяющих C++. Текущая реализация этой библиотеки работает на платформах POSIX, Win32 и Macintosh Carbon. Для непрерывных систем был использован интервальный метод Рунге–Кутта 4-го порядка, разработанный Ю. И. Шокиным [13].

Также в этой главе обсуждаются особенности реализаций алгоритмов, основанных на многомерной целочисленной и целочисленной индексации. Приведено сравнение интервального метода локализации с методами работ ([15],[8]).

Алгоритм поиска ε-траекторий реализован на языке C#. Визуализация результата реализована с помощью графического компонента для .Net платформы [9]. Пользовательский интерфейс создан с помощью технологии .Net Windows Forms [20]. Используется компилятор языка $DSDL$ для платформы Microsoft .Net.

Разработанные алгоритмы компьютерного исследования динамических систем на основе методов интервального анализа и символического образа могут быть применены для исследования непрерывных и дискретных динамических систем. Они используются в курсе по компьютерному исследованию динамических систем на математическом факультете СПбГУ, а также могут быть использованы для исследования нетривиальных динамических систем и в учебном процессе.

Заключение содержит список основных результатов, полученных в работе.

Работы автора по теме диссертации

Статьи в журналах, рекомендованных ВАК:

[1] *Ампилова Н. Б., Терентьев С. В.* О Применении интервальной арифметики при численном исследовании динамических систем. // Журнал Вестник СПбГУ, серия 10, номер 4, 2009.

[2] *Терентьев С. В.* Об оптимизации реализации алгоритма локализации инвариантных множеств динамических систем. // Журнал Вестник СПбГУ, серия 10, номер 2, 2010.

Другие публикации:

[3] *Ампилова Н. Б., Петренко Е. И., Терентьев С. В.* Разработка и реализация компьютерно–ориентированных методов исследования динамических систем с использованием символического образа. // Труды научной международной конференции „Космос, астрономия и программирование“. СПб.: СПбГУ, 2008. С. 127–134.

[4] *Ампилова Н. Б. Терентьев С. В.* Применение методов интервальной арифметики к задаче построения символического образа. // Электронный журнал „Дифференциальные уравнения и процессы управления“, номер 4, 2006. http://www.neva.ru/journal/j/index.html.

[5] *Терентьев С. В.* Применение методов интервальной арифметики к задаче построения псевдотраектрий. // Труды III Международной научно–практической конф.: Современные информационные технологии / Под ред. В. А. Сухомлин.– М.: МГУ, 2008.–С. 362–368.

[6] *Терентьев С. В.* Разработка и реализация программного комплекса для компьютерного моделирования динамических систем. // Журнал „Новые технологии“ Воронеж ВГПУ, 2008, номер 3. С. 12.

[7] *Терентьев С. В.* Разработка и реализация программного комплекса для компьютерного моделирования и исследования динамических систем. // Труды научно-технической конференции „Научное программное обечпечение в образовании и научных исследованиях“. СПб.:Изд. Санкт-Петерб. Политехн. Ун-та. 2008, январь. С. 300–306.

[8] *Терентьев С. В.* О применении интервальной арифметики при построении псевдотраекторий // Процессы управления и устойчивость: Труды 41-й международной научной конференции аспирантов и студентов / Под ред. Н. В. Смирнова, Г. Ш. Тамасяна.– СПб: Издат. СПбГУ, 2009.Ч С. 506–512.

[9] *Terentyev S.* Invariant Sets of Dynamical Systems — the Computation by Methods of Interval Arithmetic // „OVIDIUS“ UNIVERSITY ANNALS — CONSTANTZA. Series: CIVIL ENGINEERING. SPECIAL ISSUE DEDICATED TO THE 5–TH CONFERENCE п„DYNAMICAL SYSTEMS AND APPLICATIONS“. 2009. P. 219.

Цитированная литература

[1] Ампилова Н. Б., Петренко Е. И., Терентьев С. В. Разработка и реализация компьютерно–ориентированных методов исследования динамических систем с использованием символического образа. // Труды научной международной конференции „Космос, астрономия и программирование“. СПб.: СПбГУ, 2008. С. 127—134.

[2] Ахо А., Сети Р., Ульман Дж. Компиляторы: принципы, технологии, инструменты. М.: Изд. дом „Вильямс“, 2003. 768 с.

[3] Ахо А. В., Хопкрофт Д. Э. и Ульман Д. Д. Структуры данных и алгоритмы. // М. Вильямс, 2007. с.400.

[4] Осипенко Г. С. О символическом образе динамической системы. //Граничные задачи. Под ред. В. А. Алексеева, Пермь. 1983. С. 101-105.

[5] Осипенко Г. С., Ампилова Н. Б. Введение в символический анализ динамических систем. СПб.: Изд-во С.- Петербургского ун-та, 2005. 236 с.

[6] Осипенко Г. С., Романовский И. В., Петренко Е. И., Ампилова Н. Б. О вычислении спектра Морса. // Проблемы математического анализа, 2004, январь. vol. 27. с. 151-169.

[7] Графический инструмент GNUPLOT. — http://gnuplot.sourceforge.net/.

[8] Петренко Е.И. Разработка и реализация алгоритмов построения символического образа. // Электронный журнал „Дифференциальные уравнения и процессы управления„, номер 3, 2006. http://www.neva.ru/journal/j/index.html.

[9] Раба Н.О. Графические компоненты для платформы .NET. // Труды 3-й Международной научно-практической конференции „Современные информационные технологии и IT- образование". http://2008.it-edu.ru/pages/Conference-works.

[10] Романовский И. В. Дискретный анализ (учебное пособие для студентов ВУЗов) // СПб: BHV, 2003. С. 336.

[11] Смирнов А. и Флегонтов А. В. Анализ эффективности параллельных вычислений для динамических систем второго порядка. // Материалы IX Санкт-Петербургской Международной конференции „Региональная информатика-2004„. Санкт-Петербург, 2004. с. 60–61.

[12] Собрание библиотек BOOST. — http://www.boost.org.

[13] Шокин Ю. И. Интервальный анализ. Новосибирск: Наука, 1981. 111 с.

[14] Ershov A. Mixed calculation // Journal:In scientific world. 14.02.1984. http://ershov.iis.nsk.su/archive/eaindex.asp?did=2596.

[15] D. Fundinger. Investigating Dynamics by Symbolic Analysis: Tunings for an Efficient Computation of the Symbolic Image. // Электронный журнал „Дифференциальные уравнения и процессы управления", номер 3, 2005. http://www.neva.ru/journal/j/index.html.

[16] Dellnitz M. Set Oriented Methods for Dynamical Systems. // Berlin, Germany, Freie Universität Berlin, Institut für Mathematik I. 2002, Feb. vol. 2. p. 1098.

[17] Dellnitz M. and Hohmann A. A Subdivision Algorithm for the Computation of Unstable Manifolds and Global Attractors. // Numerische Mathematik, 1997. vol. 75, no. 3. p. 293–317.

[18] Dellnitz M. and Junge O. An adaptive subdivision technique for the approximation of attractors and invariant measures // Comput. Visual. Sci., 1998. no 1. p. 63–68.

[19] Matiyasevich D. Yu. and Petrenko E. I. Algorithms for the construction of isolated invariant subsets of the symbolic image // Proceedimgs of XXXVI conference „Control Processes and Stability„, St.Petersburg, 2005. p. 341–347.

[20] Microsoft Developers Network. — http://msdn.microsoft.com/library/.

[21] Minimalist GNU for Windows. — http://www.mingw.org/.

[22] Osipenko G. S. and Romanovsky J. V. and Petrenko E. I. and Ampilova N. B. Computation of the Morse Spectrum. // J. of Math. Sci., 2004. vol. 120, no. 2. p. 1155–1166. http://www.ingentaconnect.com/content/klu/joth/2004/00000120/00000002/00484193.

[23] Osipenko G. S. Dynamical Systems, Graphs, and Algorithms — Springer, 2007 Vol. 1889 of Lecture Notes in Mathematics. p. 288. http://www.springer.com/math/analysis/book/978-3-540-35593-9.

[24] Osipenko G. S. Numerical Explorations of the Ikeda mapping dynamics. // Electronic Journal of Differentional Equations and Control Processes, 2004, no. 2. p. 43–67, http://www.math.spbu.ru/diffjournal/j.