

Крымский научный центр НАН Украины

Таврический национальный университет
им. В.И. Вернадского

Черноморский филиал
Московского государственного университета

**ТАВРИЧЕСКАЯ НАУЧНАЯ КОНФЕРЕНЦИЯ
СТУДЕНТОВ И МОЛОДЫХ СПЕЦИАЛИСТОВ
ПО ИНФОРМАТИКЕ И МАТЕМАТИКЕ**

22-25 апреля 2008

КНЦ НАНУ
Симферополь

Редакционный совет:

Марянин Б. Д., к.ф.-м.н., доцент, зам. декана по научной работе ФМИ
ТНУ, ответственный редактор

Донской В. И., д.ф.-м.н., профессор, гл. научный сотрудник КНЦ НАН
Украины

Копачевский Н. Д., профессор

Чехов В. Н., профессор

Орлов И. В., доцент

Блыщик В. Ф., доцент

Анафиев А. С., доцент

Старков П. А., доцент, зам. отв. редактора

Техническая группа:

Войтицкий В. И., аспирант кафедры мат. анализа

Буряк В. В., студент

Беляев А. И., студент

Менумерова С. Р., студентка

СЕКЦИИ НАУЧНОЙ КОНФЕРЕНЦИИ:

1. МАТЕМАТИКА, председатель Копачевский Н. Д.

математический анализ, дифференциальные уравнения, алгебра, геометрия

2. ПРИКЛАДНАЯ МАТЕМАТИКА, председатель Чехов В. А.

математические модели в естествознании, численные методы решения
математических задач

3. ИНФОРМАТИКА, председатель Донской В. И.

прикладная информатика, программное обеспечение компьютерных систем

Печатается по решению научно-технического Совета КНЦ НАН и
Министерства образования и науки Украины от ... апреля 2008 года,
протокол ...

подписано к печати ...

отпечатано в типографии КНЦ НАНУ

Поиск объектов на изображении с помощью алгоритмов машинного обучения

Асанов Р. Р.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

Машинное зрение – это теория и технология создания методов, способных моделировать зрение некоторого биологического объекта. В биологии изучается зрительное восприятие человека и различных животных, в результате чего создаются модели работы таких систем в терминах физиологических процессов. Машинное зрение, с другой стороны, изучает и описывает системы машинного зрения, которые выполнены аппаратно или программно [1]. Основная цель машинного зрения – принятие решений о реальных физических объектах и сценах, основываясь на воспринимаемых изображениях. Любая система машинного зрения, основанная на машинном обучении, концептуально состоит из двух частей: настройки модели (например, классификатора), которая работает один раз, в результате получается модель с настроенными параметрами (например, настроенный классификатор), и рабочего модуля – предполагает многократное использование системы по настроенному классификатору.

В теме дипломной работы рассматривается система обнаружения объектов на изображении на основе алгоритма, предложенного в работе Виолы и Джонса [2] по поиску лиц. Алгоритм представляет собой поиск объектов при помощи скользящего окна с использованием простой схемы каскадов. На первой ступени каскада отбрасываются все положения скользящего окна, при которых в окно помещается только черный фон, или часть объекта. На вторую ступень попадают такие положения скользящего окна, при которых в окно помещается объект целиком, или значительная его часть. На второй ступени работает настроенный классификатор, который умеет отличать нужные объекты от ненужных.

Первый этап алгоритма из работы Виолы и Джонса – это построение интегрального изображения из бинарного изображения [3]. Бинаризация исходного изображения осуществляется по заранее заданному порогу. На следующем этапе работает уже настроенный классификатор на основе алгоритма AdaBoost [2, 3].

Суть алгоритма AdaBoost такова:

1) На каждой итерации мы используем некоторый метод обучения, так чтобы он выдал гипотезу которая минимизирует ошибку с учётом «важности» примеров;

2) Мы выбираем тот классификатор, который на данном шаге сделал наименьшую ошибку, с учётом «важности» прецедентов (распределение);

3) Составляем линейную комбинацию всех отобранных классификаторов.

Настройка классификатора осуществляется по базе изображений. Для этого из базы фотографий объектов нужно получить обучающую выборку, в которой записаны вектора признаков фотографий из базы и метка класса для каждого вектора (+1 для нужных объектов и -1 для ненужных объектов). По обучающей выборке нужно настроить классификатор – нужно применить алгоритм AdaBoost . Для выбора количества итераций «бустинга» нужно использовать методы эмпирической оценки качества работы классификатора (оценку на тестовой выборке или скользящий контроль) [4].

Среди преимуществ алгоритма AdaBoost можно отметить: универсальность, высокая скорость сходимости, высокая обобщающая способность, возможность очень эффективной программной реализации и распараллеливания, простота метода и отсутствия параметров. Существенным недостатком является трудность определения нужного числа итераций обучения.

ЛИТЕРАТУРА

- [1] Википедия - свободная энциклопедия.
http://ru.wikipedia.org/wiki/Компьютерное_зрение.
- [2] Paul Viola, Michael J. Jones. *Robust Real-Time Face Detection*.
<http://lear.inrialpes.fr/people/triggs/student/vj/viola-ijcv04.pdf>.
- [3] Курсы лаборатории компьютерной графики.
<http://courses.graphicon.ru/>.
- [4] Распознавание. Скользящий контроль.
<http://www.machinelearning.ru/wiki/index.php?title=CV>.
- [5] Graphics & Media lab (Лаборатория компьютерной графики при ВМК МГУ).
<http://graphics.cs.msu.ru/index.html.ru>.
- [6] Воронцов К.В. Лекции по логическим алгоритмам классификации.
<http://www.ccas.ru/voron/download/LogicAlgs.pdf.2006>.

Использование классической теории и модели Раша в теории статистической обработки тестов

Аблямитов Э. Д.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500И)

Статистический анализ теста является неотъемлемой частью тестирования. На этапе создания и верификации тестов с помощью статистического анализа проверяется валидность, надежность и точность измерений теста, внутренняя согласованность его шкал, удаляются неудачные задания, которые требуют коррекции. В автоматизированных системах контроля статистический анализ используется непосредственно в процессе тестирования для определения нужной длины теста и уровня заданий, которые целесообразно предложить ученику. После завершения тестирования с помощью аппарата математической статистики определяется интервал надежности и предлагается определенная шкала оценивания. Этим и определяется актуальность исследований по данной тематике.

Основой классической теории статистической обработки тестов является матрица тестовых результатов, в которую результаты теста заносятся дихотомически: правильный ответ - один балл, неправильный ответ - ноль баллов.

Для определения валидности теста, согласно классической теории статистической обработки, вычисляют: среднее выборочное значение, дисперсию, среднее квадратическое отклонение, асимметрию и эксцесс.

В зависимости от значения и знаков асимметрии и эксцесса закона распределения можно выделить девять таких критериев[1]:

- 1) тест является валидным, если асимметрия и эксцесс закона распределения равны нулю;
- 2) если асимметрия является отрицательной, а эксцесс положительным, то тест является невалидным;
- 3) если асимметрия и эксцесс являются отрицательными, то тест является легким, но с большой распределительной способностью;
- 4) если асимметрия является отрицательной, а эксцесс отсутствует (равен нулю), то тест является легким;
- 5) если асимметрия является положительной, а эксцесс отрицательным, то тест является сложным с большой распределительной способностью;
- 6) при положительных асимметрии и эксцессе тест является сложным с малой распределительной возможностью;

7) при положительной асимметрии и отсутствующем (равном нулю) эксцессе тест является сложным;

8) если кривая распределения симметрична, асимметрия равна нулю, а эксцесс положительный, то тест является неоднородным и сконструированным из двух субтестов: сложного и легкого;

9) если кривая распределения симметрична, асимметрия равна нулю, а эксцесс отрицательный, то тест является неоднородным, сконструированным из трех субтестов: легкого, среднего и сложного.

Помимо классической теории, для анализа результатов тестирования, современная теория предлагает логистическую модель Раша. Наиболее целесообразным считают использование модели Раша со следующей поправкой:

$$p = \frac{1}{1 + e^{-1.7(\theta - \delta)}} \quad (1)$$

Здесь p - вероятность того, что участник тестирования с уровнем подготовленности θ правильно выполнит задание с уровнем трудности δ .

Для анализа результатов тестирования реализован программный модуль, позволяющий объективно и надежно анализировать результаты тестов, как при помощи классической теории тестирования, так и при помощи модели Раша. В этом заключалась основная цель работы.

Программный модуль принимает информацию из внешнего файла или базы данных в формате таблицы Excel, предоставляет возможность пользователю просматривать исходные данные, обрабатывать полученную информацию в соответствии с выбранным методом и выдает на экран результат в виде графиков или таблиц. Кроме того на экран выводятся качественные результаты анализа.

ЛІТЕРАТУРА

- [1] Булах І.Є., Мруга М.П. *Створюємо якісний тест: Навч. посіб.* – Київ: майстер-клас, 2006. - 160 с.

Введение в технологию Hibernate и ее использование при создании Java приложений

Агафонова М.А.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. Вернадского
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

Разработка приложений для реляционных баз данных на основе объектно-ориентированных языков программирования является громоздким и трудоемким процессом. Обычно работа над автоматизацией процесса хранения данных очень скучна, и при ручной реализации существует опасность возникновения ошибок. Если к этому прибавить постоянно меняющиеся требования, разработчику необходимо учитывать сложный процесс синхронизации исходного кода и структуры хранения данных. А плюс к этому необходима переносимость приложений между платформами, и все становится еще более сложным и запутанным. В таких случаях использование технологии Hibernate имеет явные преимущества.

Hibernate object-relational mapping - решение для языка программирования Java. Оно является свободным программным обеспечением с открытым исходным кодом, которое распространяется по условиям GNU Lesser General Public License. Данное решение предоставляет легкий в использовании фреймворк для отображения объектно-ориентированной модели данных в традиционную реляционную базу данных [1].

Hibernate является программной реализацией технологии ORM (object-relational mapping). В объектно-ориентированном программировании объекты в программе представляют объекты из реального мира. Суть проблемы состоит в преобразовании таких объектов в форму, в которой они могут быть сохранены в файлах или базах данных, и которые легко могут быть извлечены в последующем, с сохранением свойств объектов и отношений между ними. Эта проблема решается следующим образом, для объектов, которые должны быть представлены в базе данных, создаются соответствующие конфигурационные файлы, в которых определяют, каким образом Java-объекты отображаются в базе данных. В этих файлах мы описываем, в какой таблице нашей базы данные будут записаны, какое поле в какой столбец таблицы будет отображено, определяются связи и отношения между объектами. Hibernate позволяет моделировать наследование, связи (один к одному, один ко многим, включение) и композицию.

Для реализации запросов к базе данных Hibernate предусматривает язык запросов, который называется Hibernate Query Language, схожий с SQL. HQL

достаточно прост в обращении. Вы найдете все знакомые ключевые слова, известные вам по SQL, такие как: SELECT, FROM и WHERE. HQL отличается от SQL тем, что вы не пишете запросы прямо в вашей модели данных (т.е. в таблицах, столбцах и т.д.), а в Java-объектах, используя их свойства и взаимосвязи. Для реализации запросов помимо языка HQL можно использовать встроенные объектные критерии во время исполнения.

Еще одним важным преимуществом этой технологии является достаточно простой способ переносимости приложений на другие платформы. Для этого достаточно внести коррективы в конфигурационный файл `hibernate.properties`, а исходный код приложения оставить без изменений. Этот файл определяет, какую базу данных мы хотим использовать, имя пользователя и пароль, и множество других опций.

Цель Hibernate состоит в том, чтобы освободить разработчика от рутинного связывания базы данных с приложением и ускорить разработку программ [2].

ЛИТЕРАТУРА

- [1] Wikipedia.Hibernate. <http://en.wikipedia.org/wiki/Hibernate>
- [2] Компьютерная газета. Введение в Hibernate и основы ORM. <http://msk.nestor.minsk.by/Kg/2005/32/kg53209.html>

Электронная цифровая подпись на основе алгоритма подобного RSA

Белый М.О.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (500и)

ОБЩИЕ ПОЛОЖЕНИЯ

При ведении деловой переписки, при заключении контрактов подпись ответственного лица является непременным атрибутом документа, преследующим несколько целей: гарантирование истинности письма путем сличения подписи с имеющимся образцом; гарантирование авторства документа. Выполнение данных требований основывается на следующих свойствах подписи: подпись аутентична, то есть с ее помощью получателю документа можно доказать, что она принадлежит подписывающему; подпись не подделываемая, то есть служит доказательством, что только тот человек, чей автограф стоит на документе, мог подписать данный документ. Подпись непереносима, то есть является частью документа и поэтому перенести ее на другой документ невозможно. Документ с подписью является неизменяемым. Любое лицо, владеющее образцом подписи может удостовериться, что документ подписан владельцем подписи. Развитие современных средств безбумажного документооборота, средств электронных платежей немислимо без развития средств доказательства подлинности и целостности документа. Таким средством является электронно-цифровая подпись (ЭЦП), которая сохранила основные свойства обычной подписи. Существует несколько методов построения ЭЦП, а именно: шифрование электронного документа (ЭД) на основе симметричных алгоритмов. Данная схема предусматривает наличие в системе третьего лица - арбитра, пользующегося доверием обеих сторон. Авторизацией документа в данной схеме является сам факт шифрования ЭД секретным ключом и передача его арбитру. Использование ассиметричных алгоритмов шифрования. Фактом подписания документа является шифрование его на секретном ключе отправителя.

СХЕМА ЭЛЕКТРОННОЙ ПОДПИСИ.

В основе каждой криптосистемы с открытым ключом лежит однонаправленная функция. Введем обозначение $E_a(X)$ - однонаправленная шифрующая функция криптосистемы, разработанной участником А. Обозначим через

$D_a(X)$ дешифрующую функцию. Согласно идеологии асимметричного шифрования функция $E_a(X)$ является общеизвестной, а $D_a(X)$ известна только А и является идентифицирующей его личность. Общая схема электронной подписи с открытым ключом выглядит так: Если участник протокола А подписывает сообщение для В, то А создает свою схему ($E_a(X)$ и $D_a(X)$). Пусть М-это сообщение. Общий вид подписи - это пара М и $D_a(X)$. Процесс проверки - это применение В функции $E_a(X)$ к аргументу $D_a(X)$ ($E_a(D_a(M))$), в результате чего должно получиться М. Таким образом криптосистему можно использовать, если $E_a(D_a(X)) = X$.

Используя общую схему для криптосистемы типа RSA, рассматриваемой в этой статье, построим протокол создания электронной подписи.

1. Генерация ключей

Выбираются два простых числа Р и Q, которые должны быть также взаимно простыми с $(P - 1)$ и $(Q - 1)$; $N = PQ$ - открытый ключ; P' и Q' - секретные ключи, такие, что $PP' = 1(mod(Q - 1))$ и $QQ' = 1(mod(P - 1))$.

2. Шифрование и дешифрование

Сторона В шифрует сообщение как $= M^N(modN)$. Сторона А находит М с помощью решения: $M = C^{p'}(modQ)$ и $M = C^{q'}(modP)$.

На основе этого, электронная подпись данной системы представлена следующей функцией:

$$D_a(X) = (P * P^{-1}mod(Q) * X^{p'}mod(Q) + Q * Q^{-1}mod(P) * X^{q'}mod(P))mod(N).$$

Проверить эту подпись может сторона А, которая получает сообщение и подпись к нему, применив шифрующую функцию, т.е.

$$E_M(D_a(M)) = E_M((P * P^{-1}mod(Q) * X^{p'}mod(Q) + Q * Q^{-1}mod(P) * X^{q'}mod(P))mod(N)) = M$$

Создание этого метода шифрования имеет актуальное значение в современной системе обработки информации, позволяет гарантировано обеспечить конфиденциальность информации, оперативно подтвердить ее достоверность, избежать арбитража третьих лиц, легко анализируется и передается современными компьютерными системами, т.е. соответствует современным требованиям обеспечения безопасности информационных систем.

Разработка приложений баз данных недвижимости на основе сетевых моделей

Бернацкий А.В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

В настоящее время в связи с быстрым изменением рынка недвижимости особо остро встает проблема быстрого поиска подходящих объектов и предоставления расширенной информации по ним. В статье рассматриваются вопросы разработки программного обеспечения на примере создания системы "Маклер".

Вначале предполагается решить задачи: изучить способы создания и разработки сетевых приложений, изучить сетевые модели, разработать структуру поисковой системы, выбрать наиболее подходящие приёмы и инструментальные средства, операционную систему и серверное программное обеспечение для решения данной задачи, а также использовать следующие методы исследования: изучить и проанализировать научную и методическую литературу по вопросам программирования приложений и баз данных, апробировать результаты работы на базе данных большого размера.

Разрабатываемый программный продукт позволит автоматизировать работу сотрудника агентства недвижимости, ускорить рабочие процессы, тем самым увеличив прибыль компании. В связи с этим к проекту предъявляются следующие требования:

- высокая скорость доступа к базе данных;
- удобство поиска и редактирования;
- защищенность;
- удобный и быстрый в использовании интерфейс;
- получение сводных отчетов.

Программное обеспечение в первую очередь должно реализовать следующие функции:

- поиск информации об объектах недвижимости по ключевым словам;
- расширенный поиск информации по любым параметрам;
- создание электронных справочников;
- поддержка операций "продажа", "покупка", "обмен" и т.д.;
- учет заявок (неограниченное количество реквизитов для описания объекта недвижимости);
- ведение истории заявки;
- создание произвольных форм отчетов;

- экспорт данных в форматы распространенные форматы данных (Word, Excel, HTML, PlainText);
- поддержка одновременной работы неограниченного числа пользователей;
- мощная и быстрая база данных;
- удобный и простой интерфейс.

Для разработки программного продукта использовались следующие средства:

- для программирования модулей системы выбран язык объектно-ориентированного программирования Delphi;
- для хранения часто изменяемой информации выбран SQL-сервер InterBase;
- в качестве методики программирования выбрано объектно-ориентированное программирование.

База данных включает 11 функциональных таблиц, предназначенных для хранения информации об объектах недвижимости и их характеристиках. Для каждой таблицы определены соответствующие атрибуты (поля). Во всех таблицах для каждого атрибута выбран соответствующий тип данных и оптимальный размер хранимых в этих полях данных.

Разработка логической структуры базы данных также включает в себя определение и создание индексов для ключевых и неключевых полей таблиц, определение и создание связей между таблицами, необходимых для работы запросов. В итоге получена законченная, достаточно сложная логическая структура базы данных, состоящая из большого количества таблиц, связанных между собой особым образом.

Для данной структуры базы данных, разработаны структура и функциональные модули системы, получен интерфейс пользователя, позволяющие пользователю работать с базой данных. Проведена апробация базы данных с помощью отладочной информации.

В настоящее время разрабатываются алгоритмы приложений, основанные на сетевых моделях: задача коммивояжера для определения оптимального маршрута объектов недвижимости, задача нахождения кратчайшего пути. Исходные данные извлекаются из базы данных. Это существенно дополнит функции системы по сравнению с обычной базой данных и сделает ее уникальной.

Решение задачи оптимального быстрогодействия с помощью метода деформируемого многогранника

Бирдус Л. Г.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ (ГРУППА 500П)

Рассмотрим *постановку задачи* оптимального быстрогодействия и решение этой задачи с помощью метода деформируемого многогранника (метод Нелдера-Мида)[3]:

Постановка 1. Движение объекта управления в плоскости описывается следующей системой дифференциальных уравнений[1]:

$$\begin{aligned}x'' + ax' &= U_x(t), \\ y'' + ay' &= U_y(t),\end{aligned}\tag{1}$$

где $x'(t)$, $x''(t)$, $y'(t)$, $y''(t)$ - производные координат по времени, $U_x(t)$, $U_y(t)$ - составляющие управления $U(t)$, a - заданная константа. Управление ограничено:

$$\|U(t)\| \leq 1\tag{2}$$

Начальные координаты и скорости объекта заданы

$$\begin{aligned}x(0) = x_0, x'(0) &= V_{x_0} \\ y(0) = y_0, y'(0) &= V_{y_0}\end{aligned}\tag{3}$$

Необходимо найти управление $U(t)$, приводящее объект (1) из заданного начального положения (3) в заданное конечное положение

$$\begin{aligned}x(T) = 0, x'(T) &= 0 \\ y(T) = 0, y'(T) &= 0\end{aligned}\tag{4}$$

за минимальное время T .

Для этого сведем систему из двух дифференциальных уравнений второго порядка к системе четырех дифференциальных уравнений первого порядка:

$$\begin{aligned}x' &= V_x, \\ V_x &= -aV_x + U_x, \\ y' &= V_y, \\ V_y &= -aV_y + U_y\end{aligned}\tag{5}$$

Далее воспользуемся принципом максимума Понтрягина и получим краевую задачу:

$$\begin{aligned}
 p_1' &= 0, \\
 p_2' &= -(p_1 - ap_2), \\
 p_3' &= 0, \\
 p_4' &= -(p_3 - ap_4), \\
 x' &= V_x, \\
 V_x' &= -aV_x + \frac{p_2}{\sqrt{p_2^2 + p_4^2}}, \\
 y' &= V_y, \\
 V_y' &= -aV_y + \frac{p_4}{\sqrt{p_2^2 + p_4^2}}
 \end{aligned} \tag{6}$$

где p_1, p_2, p_3, p_4 - коэффициенты функции Понтрягина[2]. Добавим еще четыре начальных условия для сопряженных переменных:

$$p_1(0) = p_{1,0} \quad p_2(0) = p_{2,0} \quad p_1'(0) = V_{p_{1,0}} \quad p_2'(0) = V_{p_{2,0}} \tag{7}$$

Получаем задачу Коши (6), (3), (7), которую будем численно решать, задав некоторое время T .

В результате численного интегрирования системы дифференциальных уравнений (6), (3), (7) получим фазовые траектории, которые, для произвольно заданных условий (7) и времени T , не проходят через начало координат. Чтобы найти решение, удовлетворяющее условию

$$x^2(T) + y^2(T) + V_x^2(T) + V_y^2(T) \rightarrow 0 \tag{8}$$

нужно решить задачу минимизации функции пяти переменных $\Phi(p_{1,0}, p_{2,0}, V_{p_{1,0}}, V_{p_{2,0}}, T)$, значения которой рассчитываются путем решения задачи Коши (6), (3), (7) и последующего вычисления значения выражения (8). Эту задачу мы будем решать с помощью метода Нелдора-Мида.

Описание метода

Метод заключается в построении и последовательном преобразовании построенного многогранника в пространстве параметров[4].

Шаг 1. Строим многогранник с $n + 1$ вершиной. Для этого задаем первый вектор $\vec{x}_1 \subseteq X^n$. Остальные n векторов строим по формулам $\vec{x}_i = \vec{x}_1 + k\vec{e}_i$, где $\vec{e}_i, i = \overline{1, n}$

Далее находим значения минимизируемой функции в вершинах многогранника, а также минимальное Φ_l , максимальное Φ_h значения функции $\Phi(x_i)$, и значение перед максимальным Φ_g и запоминаем соответствующие вершины x_l, x_h, x_g .

Шаг 2. Найдем центр тяжести $x_c = \frac{1}{n+1} \sum x_i, i = \overline{1, n}$.

Приступим к деформации многогранника: возьмем $x_r = x_c + \alpha(x_c - x_h)$ и сравним значения $\Phi(x_l)$ и $\Phi(x_r)$.

Если $\Phi(x_h) < \Phi(x_r)$, то возьмем $x_h = x_c + \gamma(x_r - x_c)$.

Теперь будем сравнивать значения функций $\Phi(x_r)$ и $\Phi(x_e)$. Если $\Phi(x_r) > \Phi(x_e)$, то x_h делаем равным x_e , иначе $x_h = x_c$. Получаем новый многогранник, который далее и будем деформировать

Шаг 3. В полученном многограннике проверяем будет ли $\max \Phi(x_i) > \Phi(x_c), i = \overline{1, n}$. Если да, то заменяем x_h на x_c и получаем новый многогранник.

Далее возьмем x_q , такое что $\Phi(x_q) = \min(\Phi(x_r), \Phi(x_e))$. Возьмем $x_s = x_c + \beta(x_q - x_c)$. Если $\Phi(x_s) > \Phi(x_q)$, то заменяем все вершины многогранника x_i на $x_i = x_i + \frac{1}{2}(x_l - x_c)$ и продолжаем деформацию полученного многогранника. Если $\Phi(x_s) < \Phi(x_q)$, то x_h заменяем x_s . И работаем с полученным новым множеством вершин.

Замечание 1. Достижение оптимального решения определяется условием: $\sum (f_i - \bar{f})/n + 1$, где $\bar{f} = \sum f_i/n + 1$ [4]

Замечание 2. Коэффициенты сжатия $\beta > 1$, растяжения $\gamma > 1$ и отражения $\alpha > 0$

С помощью метода деформируемого многогранника находим оптимальные значения параметров T, p_1, p_2, p_3, p_4 строим фазовые траектории решения задачи (1), (3), (4).

Пример 1. Построены фазовые траектории для начальных значений $x(0) = 2, y(0) = 2, V_x(0) = 1, V_y(0) = 1$ и значений параметра $a = 0.5, 1, 2$ Фазовые траектории показаны на рисунке 1.

ЛИТЕРАТУРА

- [1] Евтушенко Ю.Г. Методы решения экстремальных задач и их применение в системах оптимизации//М.:Наука,1982.
- [2] Васильев Ф.П. Численные методы решения экстремальных задач//М.:Наука,1988.

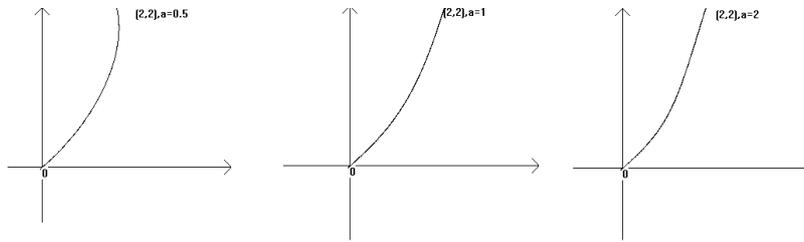


Рис. 1. Фазовые траектории

- [3] Козлов В.И. Численное решение задачи о мягкой посадке в теоретико-игровой постановке// Ученые записки Таврического Национального Университета, Вып.7(46), С.:2001
- [4] Губарь Ю.В. Введение в математическое программирование//интернет-университет Intuit, www.intuit.ru

О классах интегральных уравнений заданных на полуоси

Бойко А.П.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. Вернадского
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА АЛГЕБРЫ И ФУНКЦИОНАЛЬНОГО АНАЛИЗА (ГРУППА 500)

В статье рассматриваются различные классы интегральных уравнений на полуоси - примеры интегрального уравнения Фредгольма первого рода на полуоси.

Первые примеры интегральных уравнений появились в работах Фурье, Абеля, Лиувилля. После решения Нейманом в 1877 году задачи Дирихле для уравнения Лапласа путем сведения ее к интегральному уравнению, а также с появлением работ Вольтерра, такие задачи стали важнейшим математическими объектами для изучения.

Рассмотрим примеры наиболее распространенных интегральных уравнений на полуоси.

Пример 1. Существует широкий класс задач Винера-Хопфа, возникших в связи с исследованиями радиационного равновесия звезд. Уравнение вида

$$W\psi(x) := c\psi(x) + \int_0^\infty k(x-t)\psi(t) dt = f(t), \quad x \in (0, \infty)$$

называется уравнением Винера-Хопфа первого рода при $c = 0$, второго рода при $c \neq 0$. Для решения этих уравнений разработаны многочисленные устойчивые методы [1].

Пример 2. Известная задача Зоммерфельда о диффракции на полуплоскости с помощью функции Грина сводится к решению интегрального уравнения вида

$$i \int_0^\infty H_0^{(1)}(k|x-\xi|)h(\xi) d\xi = 2e^{-ikx\cos\theta},$$

где k имеет положительную мнимую часть, $H_0^{(1)}(x) = J_0(x) + iY_0(x)$ - функция Ханкеля, а $J_0(x)$ и $Y_0(x)$ - функции Бесселя первого и второго рода соответственно [1].

Пример 3. Из многочисленных прикладных задач известно интегральное уравнение Фредгольма первого рода

$$Kx(t) := \int_0^\infty k(t,\tau)b(\tau)x(\tau) d\tau = y(t), \quad t \geq 0, \quad (1)$$

где $x(t), y(t) \in L_2(0, \infty)$, а $k(t, \tau), b(\tau)$ - непрерывные функции, такие что

$$|k(t, \tau)| \leq \frac{c_k}{[(1+t)(1+\tau)]^\kappa}, \quad |b(\tau)| \leq c_b\tau^\beta,$$

$$t, \tau \in [0, \infty), \quad c_k, c_b > 0. \quad (2)$$

Если оператор K компактный в рассматриваемом функциональном пространстве, то задача (1) некорректно поставлена и для построения приближенного метода ее решения требуется применение специальных регуляризующих алгоритмов [2].

Пример 4. На интервале $(0, \infty)$ рассмотрим прямое и обратное преобразования Ханкеля вида

$$\begin{aligned} \Phi(u) &= \int_0^\infty f(t)tJ_\nu(ut) dt, \quad (0 < u < +\infty), \\ f(x) &= \int_0^\infty \Phi(u)uJ_\nu(ut) du, \quad (0 < t < +\infty), \quad \nu \geq 1, \end{aligned}$$

где $J_\nu(x)$ - функция Бесселя порядка ν . Для достаточно больших u имеет место асимптотическое представление

$$J_\nu(u) \sim \sqrt{\frac{2}{\pi u}} \cos\left(u - \frac{\pi}{2}\nu - \frac{\pi}{4}\right), \quad (3)$$

а значит условия (2) выполняются при $k = 1/2$, $\beta = 1$.

Пусть J – неограниченный самосопряженный положительно определенный оператор в $L_2(0, \infty)$ вида $Jf(t) = (1+t)^{1/2}f(t)$. Через $L_{2,s}$ обозначим гильбертово пространство, порождаемое оператором J^s , со скалярным произведением $\langle f, g \rangle_s = \langle J^s f, J^s g \rangle$ и соответствующей нормой $\|f\|_s = \langle f, f \rangle_s^{1/2}$. Очевидно, что $\|f\|_0 = \|f\|_{L_2(0, \infty)}$. Далее рассмотрим уравнение (1) при условии, что $x \in L_{2,s}$ тогда в силу предположения (2) выполняется

$$\begin{aligned} \|Kx\|^2 &\leq c \int_0^\infty (1+t)^{-2k} dt \left(\int_0^\infty (1+\tau)^{\beta-k-\frac{s}{2}} x(\tau)(1+\tau)^{\frac{s}{2}} d\tau \right)^2 \leq \\ &\leq c \int_0^\infty (1+t)^{-2k} dt \int_0^\infty (1+\tau)^{2\beta-2k-s} d\tau \|x\|_s^2, \quad c > 0 \end{aligned}$$

Значения интегралов в правой части последнего неравенства будут ограничены, если

$$k > 1/2, \quad s > 2\beta - 2k + 1. \quad (4)$$

Таким образом, при ограничениях (4) оператор K уравнения (1) непрерывно действует из пространства $L_{2,s}$ в $L_{2,0} = L(0, \infty)$.

Пример 5. Рассмотрим уравнение, описывающее распределение размера сферических частиц методом рассеивания:

$$\int_0^\infty x(\tau)\tau^4 \left[\frac{2J_1(t\tau)}{t\tau} \right]^2 d\tau = y(t), \quad (5)$$

где J_1 - функция Бесселя 1-го порядка и x - неизвестное распределение размера частицы. Это уравнение является примером задачи (1) при условии (4). Из (3) следует, что для уравнения (5) $b(\tau) = \tau^4$ и

$$k(t, \tau) = \left[\frac{2J_1(t\tau)}{t\tau} \right]^2 \sim \frac{8}{\pi} (t\tau)^{-3} \sin^2\left(t\tau - \frac{\pi}{4}\right).$$

Что означает: условия (2) выполнены при $k = 3$ и $\beta = 4$. В свою очередь, для выполнения (4) достаточно взять $s > 3$.

Рассматривая задачу приближенного решения интегрального уравнения Фредгольма на полусоси, можно установить справедливость оценки $\|x - x_\delta\| = O(\delta^\xi)$, где x_δ -приближенное решение уравнения (1).

ЛИТЕРАТУРА

- [1] *А.В. Манжиров, А. Д. Полянин* Справочник по интегральным уравнениям // Методы решения . Москва 2000г.
- [2] *А.Н. Тихонов, А.В. Гончарский, В.В. Степанов, А.Г. Ягола* Регуляризирующие алгоритмы и априорная информация // Москва 1983г.

Использование метода интерполирования по Лагранжу для построения векторно-параметрической кривой

Голыцына Н. А.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ (ГРУППА 500-П)

Целью работы является построение векторно-параметрической кривой, представленной многочленом n -ой степени, и изучение ее свойств.

Рассмотрим метод интерполирования по Лагранжу, чтобы получить представление векторно-параметрической кривой, задаваемой полиномом третьей степени [1].

Выбираем в качестве интерполируемых точки, определяемые значениями $u = 0$ $r = r_0$, $u = 1/3$ $r = r_1$, $u = 2/3$ $r = r_2$, $u = 1$ $r = r_3$.

Тогда, используя классическое представление интерполяционного полинома Лагранжа, получим:

$$r(u) = \frac{9}{2} [r_0(1-u)\left(\frac{2}{3}-u\right)\left(\frac{1}{3}-u\right) + 3r_1(1-u)\left(\frac{2}{3}-u\right)u + 3r_2(1-u)\left(u-\frac{1}{3}\right)u + r_3\left(u-\frac{2}{3}\right)\left(u-\frac{1}{3}\right)u], \quad (1)$$

где r_0, r_1, r_2, r_3 – вектор-функции точек 0, 1, 2, 3.

Аналогично можно записать векторно-параметрическое уравнение аналогичного сегмента кривой n -ой степени для $n + 1$ точки:

$$r = \frac{n^{n-1}}{(n-1)!} \sum_{i=0}^n (-1)^i r_i \frac{n!}{(n-i)!i!} \prod_{j=0, j \neq i}^n \left(\frac{j}{n} - u\right), \quad 0 \leq u \leq 1, \quad (2)$$

где n – порядок кривой.

На практике было замечено, что интерполяционные многочлены (2) и (1), как частный случай (2), позволяют получить желаемый результат лишь в случае равномерного задания каркасных точек.

Рассмотрим теперь вопрос состыковки сегментов кривой, который необходимо решить при моделировании гладкой сплайновой кривой.

В статье [1] был предложен следующий подход к решению этой задачи.

Возьмем от (1) производные и рассмотрим два состыкованных сегмента (i) и ($i + 1$).

В сегменте (i) при $u = 1$ получим производную:

$$r'_{u=1}{}^{(i)} = -r_0^{(i)} + \frac{9}{2}r_1^{(i)} - 9r_2^{(i)} + \frac{11}{2}r_3^{(i)}. \quad (3)$$

В сегменте $(i + 1)$ при $u = 0$ получим производную:

$$r'_{u=0}{}^{(i+1)} = -\frac{11}{2}r_0^{(i+1)} + 9r_1^{(i+1)} - \frac{9}{2}r_2^{(i+1)} + r_3^{(i+1)}. \quad (4)$$

Тогда после приравнивания (3) и (4) в статье была получена следующая формула для каждой точки соединения сегментов:

$$\frac{9}{2}(r_1^{(i)} - r_2^{(i+1)}) - 9(r_2^{(i)} - r_1^{(i+1)}) = (r_0^{(i)} - r_3^{(i+1)}), \quad i = 0, \dots, n - 1, \quad (5)$$

которая, по мнению авторов, позволяла построить гладкое соединение сегментов, так как образовывалась трёхдиагональная система.

Как оказалось, в процессе расчета автором статьи была допущена ошибка и, как следствие, данная формула не верна, что можно легко доказать, рассмотрев контрпример.

Основными условиями для формирования составной кривой являются ее следующие свойства [2]:

Свойство 1. *Непрерывность составной кривой.*

Свойство 2. *Непрерывность касательного вектора по направлению в точках соединения.*

Свойство 3. *Непрерывность кривизны в точках соединения.*

Приравняв (3) и (4), с учетом свойств 1 и 2 получим:

$$\frac{9}{2}(r_1^{(i)} + r_2^{(i+1)}) - 9(r_2^{(i)} + r_1^{(i+1)}) + 11r_3^{(i)} = (r_0^{(i)} + r_3^{(i+1)}). \quad (6)$$

Продифференцировав (1) два раза, с учетом свойств 1, 2 и 3 получим:

$$4(r_2^{(i+1)} - r_1^{(i)}) + 5(r_2^{(i)} - r_1^{(i+1)}) = (r_3^{(i+1)} - r_0^{(i)}). \quad (7)$$

Результат, полученный после исправления ошибки, существенно отличается от результата, полученного в упомянутой статье. Однако, легко видеть, что при задании концевых точек сегмента, уравнения (6) и (7) образуют систему двух линейных алгебраических уравнений с двумя неизвестными. Из этой системы можно найти две промежуточные точки, и затем построить сегмент кривой, воспользовавшись уравнением (1).

Таким образом, построив один сегмент, имея концевую точку следующего сегмента, можно решить вопрос их состыковки и как результат получить гладкую сплайновую кривую, используя описанный метод.

ЛІТЕРАТУРА

- [1] Бадаев Ю.И., Ковтун О.М. *Сплайновые векторно-параметрические кривые* // Прикладная геометрия и инженерная графика. – К.:КНУБА. Вып. 72. – 2003 – С. 47-49.
- [2] Фокс А., Пратт М. *Вычислительная геометрия. Применение в проектировании и производстве.* – М.: Мир, 1982. – 304 с.

Ряды аналитических функций кватернионного переменного

Груздѣва Д.С.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА АЛГЕБРЫ И ФУНКЦИОНАЛЬНОГО РЯДА (ГРУППА 500)

В последнее время появилось много, связанных с детальным изучением спектральных задач для линейных операторов, действующих в гильбертовых кватернионных пространствах. В этой связи возникает вопрос об определении и свойствах аналитических функций кватернионного переменного.

ДИФФЕРЕНЦИРУЕМОСТЬ ФУНКЦИЙ КВАТЕРНИОННОГО ПЕРЕМЕННОГО.

Пусть \mathbb{H} — тело кватернионов. Рассмотрим функцию $\varphi : \mathbb{H} \rightarrow \mathbb{H}$ с областью определения $D(\varphi)$. Всюду в данной работе $F \supset \mathbb{R}$ — вещественное поле в \mathbb{H} , вещественным базисом которого являются кватернионы $1, f$, где $f^2 = -1$. Пусть дана невещественная точка $q_0 \in D(\varphi)$. Так как всякий невещественный кватернион однозначно определяет невещественное поле $F(q_0)$, которое содержит данный кватернион, то, не нарушая общности рассуждений, можно считать, что в этом случае имеем дело с полем $F = F(q_0)$. Кроме того, существует такой кватернион ε : $\varepsilon^2 = -1$, что кватернионы $1, f, \varepsilon, f\varepsilon$ образуют вещественный базис в \mathbb{H} , относительно которого всякий кватернион допускает разложение $q = q_1 + q_2\varepsilon$, где $q_1, q_2 \in F$. Следовательно, для всякого $q \in D(\varphi)$ функцию φ можно представить в виде $\varphi(q) = \varphi_1(q_1, q_2) + \varphi_2(q_1, q_2)\varepsilon$, где функции φ_1, φ_2 определены на некотором подмножестве F^2 и отображают это подмножество в F . Также будем говорить, что величина h стремится к нулю относительно множества $F(q_0)$ ($h \rightarrow 0 \cdot \mathbb{F}(q_0)$), если $|h|$ стремится к нулю, причем $h \in F(q_0)$.

Функцию $\varphi : \mathbb{H} \rightarrow \mathbb{H}$ назовем *дифференцируемой справа* в точке q_0 , если существует предел:

$$\lim_{h \rightarrow 0 \cdot \mathbb{F}(q_0)} (\varphi(q_0 + h) - \varphi(q_0))h^{-1}, \quad (1)$$

который будем называть правой производной функции φ в точке q_0 и обозначать $\varphi'(q_0)$.

Обозначим через $\psi_1(q_1) = \varphi_1(q_1, 0)$, $\psi_2(q_1) = \varphi_2(q_1, 0)$. Как показано в работе [1], функция $\varphi(q)$ дифференцируема в невещественной точке q_0 тогда и только тогда, когда функции $\psi_1(q)$ и $\overline{\psi_2}(q)$ дифференцируемы в этой точке в поле $F(q_0)$. Причем, $\varphi'(q_0) = \psi_1'(q_0) + \overline{\psi_2}'(q_0)\varepsilon$.

Пусть F — некоторое фиксированное поле. Функцию φ , действующую в \mathbb{H} , назовем *аналитической* в области $G \subset F$, если она дифференцируема в каждой точке этой области.

РЯДЫ АНАЛИТИЧЕСКИХ ФУНКЦИЙ КВАТЕРНИОННОГО ПЕРЕМЕННОГО.

Пусть в области $G \subset F$ определена бесконечная последовательность функций $\{\varphi_n(q)\}_{n=1}^{\infty}$. Выражение $\sum_{n=1}^{\infty} \varphi_n(q)$ будем называть функциональным рядом функций кватернионного переменного. Аналогично комплексному случаю определим сходимость (равномерную сходимость) этого ряда в области G к функции $\varphi(q)$, которую назовем суммой функционального ряда.

Теорема 1. Пусть функции $\varphi_n(q)$ являются аналитическими в G , а ряд $\sum_{n=1}^{\infty} \varphi_n(q)$ сходится равномерно в любой замкнутой подобласти $\overline{G'} \subset G$ к функции $\varphi(q)$. Тогда

1) $\varphi(q)$ — аналитическая функция в любой замкнутой подобласти $\overline{G'} \subset G$;

2) $\varphi^{(k)}(q) = \sum_{n=1}^{\infty} \varphi_n^{(k)}(q)$ и $\sum_{n=1}^{\infty} \varphi_n^{(k)}(q) \Rightarrow \varphi^{(k)}(q)$ в любой замкнутой подобласти $\overline{G'} \subset G$.

Доказательство. С учетом приведенного выше разложения для функций кватернионного переменного имеем $\varphi(q) = \psi_1(q) + \psi_2(q)\varepsilon$, $\varphi_n(q) = \psi_{1,n}(q) + \psi_{2,n}(q)\varepsilon$. Тогда нетрудно показать, что ряд $\sum_{n=1}^{\infty} \varphi_n(q)$ равномерно сходится к функции $\varphi(q)$ в том и только в том случае, когда к функциям $\psi_1(q)$, $\overline{\psi_2}(q)$ соответственно равномерно сходятся ряды $\sum_{n=1}^{\infty} \psi_{1,n}(q)$ и $\sum_{n=1}^{\infty} \overline{\psi_{2,n}}(q)$.

В силу изоморфизма поля \mathbb{C} и поля F можно утверждать [2], что функции $\psi_1(q)$ и $\overline{\psi_2}(q)$ являются аналитическими в любой замкнутой подобласти $\overline{G'} \subset G$. Следовательно, функция $\varphi(q)$ также является аналитической в $\overline{G'} \subset G$, причем справедлива следующая формула для равномерно сходящихся рядов

$$\varphi^{(k)}(q) = \sum_{n=1}^{\infty} \psi_{1,n}^{(k)}(q) + \left(\sum_{n=1}^{\infty} \overline{\psi_{2,n}^{(k)}}(q) \right) \varepsilon = \sum_{n=1}^{\infty} \varphi_n^{(k)}(q) \quad (2)$$

Таким образом, теорема доказана. \square

ЛІТЕРАТУРА

- [1] Карпенко И.И., Сухтаев А.И., Тышкевич Д.Л. *Об одном подходе к дифференцированию функций кватернионного переменного* // Ученые записки Таврического национального университета, т. 17(56), Вып. 1. — 2004. — С. 30–37
- [2] Привалов И.И. *Введение в теорию функций комплексного переменного* // М.: "Наука" Главная редакция физико-математической литературы — 1984. — 432с.

ТЕОРЕМА БАНАХА-ГРОТЕНДИКА ДЛЯ СЧЕТНЫХ ИНДУКТИВНЫХ ШКАЛ ЛОКАЛЬНО ВЫПУКЛЫХ ПРОСТРАНСТВ

Демидова М.Н.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА АЛГЕБРЫ И ФУНКЦИОНАЛЬНОГО АНАЛИЗА (ГРУППА 500)

Аннотация. Теорема Банаха-Гротендика обобщена на случай счетных индуктивных шкал ЛВП при достаточно общих условиях.

Классическая теорема Банаха-Гротендика является одним из базовых результатов теории двойственности ЛВП. Она отвечает на следующий вопрос: при каких условиях функционал $\Lambda \in E_\beta^*$ из второго сопряженного к произвольному ЛВП E принадлежит E ?

Теорема 2 (Банаха-Гротендика). $\Lambda f = \Lambda_x f = f(x)$ ($\forall f \in E^*$) при некотором $x \in E$ тогда и только тогда, когда для некоторой базы окрестностей нуля $\mathcal{U} = \{U_\alpha\}_{\alpha \in A}$ в E все сужения $\Lambda|_{U_\alpha}^0$ (на поляры $\overset{0}{U}_\alpha$ к U_α) *-слабо непрерывны.

Отметим прежде всего, что теореме [1] нетрудно придать глобальную формулировку, пользуясь разложением E^* в шкалу пространств. Пусть $\{\|\cdot\|_\alpha\}_{\alpha \in A}$ — индуктивно упорядоченная определяющая система полунорм в E . Положим для каждого $\alpha \in A$:

$$E^{*\alpha} = \{f \in E^* \mid \|f\|^\alpha = \sup_{\|x\|_\alpha \leq 1} |f(x)| < +\infty\}.$$

Пространства $E^{*\alpha}$ образуют так называемое *нормальное разложение* E^* в индуктивную шкалу векторных пространств:

$$\overrightarrow{E^*} = \{E^{*\alpha}\}_{\alpha \in A} \quad (\alpha_1 \leq \alpha_2 \Rightarrow E^{*\alpha_1} \hookrightarrow E^{*\alpha_2}).$$

Снабжая пространства $E^{*\alpha}$ *-слабыми топологиями $\sigma(E^{*\alpha}, E)$, мы построим слабое оснащение $\sigma(\overrightarrow{E^*}, E)$ шкалы $\overrightarrow{E^*}$:

$$(\overrightarrow{E^*}, \sigma(\overrightarrow{E^*}, E)) = \{(E^{*\alpha}, \sigma(E^{*\alpha}, E))\}_{\alpha \in A}.$$

Положим теперь $U_\alpha = \{x \in E \mid \|x\|_\alpha \leq 1\}$, $\alpha \in A$. Тогда $E^{*\alpha} = sp \overset{0}{U}_\alpha$, и, следовательно, сужения $\Lambda|_{U_\alpha}^0$ *-слабо непрерывны тогда и только тогда, когда сужения $\Lambda|_{E^{*\alpha}}$ непрерывны относительно топологий $\sigma(E^{*\alpha}, E)$. Это в точности означает, что Λ принадлежит сопряженному пространству к шкале $(\overrightarrow{E^*}, \sigma(\overrightarrow{E^*}, E))$. Таким образом, теорема [1] принимает следующий вид.

Теорема 3 (Банаха-Гротендика). *Имеет место векторный изоморфизм:*

$$(\overrightarrow{E^*}, \sigma(\overrightarrow{E^*}, E))^* \cong E. \quad (1)$$

Возникает естественный вопрос о переносе изоморфизма [1] на случай индуктивных шкал ЛВП \overrightarrow{E} , ответ на который составляет основное содержание работы. Введем необходимые понятия.

Определение 1. Пусть дана индуктивная шкала векторных пространств $\overrightarrow{E} = (\{E_n\}_{n \in N}, \{e_n : E_n \hookrightarrow E_{n+1}\}_{n \in N})$. Если для любого $n \in N$ в пространстве E_n задана система локально выпуклых топологий $\{r_n^m\}_{m \leq M_n}$, $M_n \leq +\infty$, причем:

- (1) для любого фиксированного $n \in N$ система $\overleftarrow{E}_n = \{(E_n^m, \hat{r}_n^m)\}_{m \leq M_n}$ — проективная шкала ЛВП. (E_n^m — фактор-пространства $E_n / \ker r_n^m$, пополненные в фактор-топологиях \hat{r}_n^m);
- (2) для любого фиксированного $m \in M$ система $\overrightarrow{E}^m = \{(E_n^m, \hat{r}_n^m)\}_{n: m \leq M_n}$ — индуктивная шкала ЛВП,

то система $r(\overrightarrow{E}) = \{r_n^m\}_{n \in N, m \leq M_n}$ называется проективным оснащением шкалы \overrightarrow{E} , пара $(\overrightarrow{E}, r(\overrightarrow{E}))$ проективно оснащенной шкалой, а топологии

$$(E_n, r(E_n)) = \varprojlim_{m \rightarrow M_n} (E_n^m, \hat{r}_n^m)$$

— топологиями, порожденными оснащением $r(\overrightarrow{E})$. Индуктивная шкала ЛВП

$$\overrightarrow{E}_r = \{(E_n, r(E_n))\}_{n=1}^{\infty}$$

называется шкалой, порожденной оснащением $r(\overrightarrow{E})$.

В частности, оснащение $\nu(\overrightarrow{E}) = \{\{\nu_n^m\}_{m \leq M_n}\}_{n \in N}$, порожденное системами полунорм $\{\|\cdot\|_n^m\}_{m \leq M_n}$ в ЛВП E_n ($\nu_n^m \sim \|\cdot\|_n^m$) назовем нормальным оснащением шкалы \overrightarrow{E} .

Определение 2. Пусть $\overrightarrow{E}_r = \{(E_n, r(E_n))\}_{n=1}^{\infty}$ — индуктивная шкала ЛВП с вложениями $e_n : E_n \hookrightarrow E_{n+1}$. Фундаментальной шкалой $\overrightarrow{\Lambda} = \{\Lambda_n\}_{n=1}^{\infty}$ на \overrightarrow{E} назовем систему функционалов $\{\Lambda_n \in (E_n, r(E_n))^*\}_{n=1}^{\infty}$ с условиями согласованности: $\Lambda_n = \Lambda_{n+1} \circ e_n$ ($n = 1, 2, \dots$).

Определение 3. Пусть $\overrightarrow{E}_r = (\overrightarrow{E}, r(\overrightarrow{E}))$ — проективно оснащенная индуктивная шкала ЛВП, $r(\overrightarrow{E}) = \{r_n^m\}_{n \in N, m \leq M_n}$, $(E_n, r(E_n)) = \varprojlim_{m \rightarrow M_n} (E_n^m, \hat{r}_n^m)$,

$M_n \leq +\infty$, $M_n \searrow$. Если $\vec{\Lambda} \in E_r^*$, то для любого $n \in N$ существует такое $m \leq M_n$, что $\Lambda_n \in (E_n^m, \hat{r}_n^m)$. Положим

$$\rho_\Lambda(n) = \min\{m \leq M_n \mid \Lambda^n \in (E_n^m, \hat{r}_n^m)^*\} \quad (n \in N).$$

Отображение ρ_Λ назовем проективным индексом $\vec{\Lambda}$ относительно оснащения $r(\vec{E})$.

Множество всех проективных индексов, порожденных оснащением $r(\vec{E})$ обозначим $P(r)$. Во множестве $P(r)$ введем поточечный порядок

$$(\rho_1 \leq \rho_2) :\Leftrightarrow (\rho_1(n) \leq \rho_2(n), \forall n \in \mathbb{N}).$$

Определение 4. Пусть \vec{E}_r — проективно оснащенная шкала векторных пространств. Для любого проективного индекса $\rho \in P(r)$ положим:

$$E^{*\rho} = \{\vec{\Lambda} \in E_r^* \mid \rho_\Lambda \leq \rho\}.$$

Разложение пространства E_r^* в индуктивную шкалу векторных пространств

$$\vec{E}_r^* = \{E^{*\rho}\}_{\rho \in P(r)}$$

назовем проективным разложением пространства E_r^* , а шкалу \vec{E}_r^* — проективно сопряженной шкалой к \vec{E}_r . В частности, если \vec{E}_ν — нормально оснащенная шкала, то шкалу \vec{E}_ν^* назовем нормальным разложением пространства E_ν^* .

С помощью введенных выше понятий аналог теорем [1],[2] для счетных индуктивных шкал ЛВП можно сформулировать следующим образом.

Теорема 4 (Обобщенная Банаха-Гротендика). Пусть \vec{E}_ν — счетная нормально оснащенная индуктивная шкала пространств Фреше, $\vec{E}_\nu = \left\{ E_n = \varprojlim_{m \rightarrow \infty} (E_m^n, \hat{r}_m^n) \right\}_{n \in N}$. Предположим, что для любого $n \in N$ и любого $m \leq M_n$:

- (1) E_m^n — рефлексивные банаховы пространства;
- (2) вложения $E_n \hookrightarrow E_m^n \hookrightarrow E^m$ плотны.

Тогда имеет место векторный изоморфизм:

$$\overrightarrow{(E_\nu^*, \sigma(E_\nu^*, \vec{E}_\nu))^*} \cong \vec{E}_\nu.$$

В доказательстве используется теорема о перестановке индуктивных и проективных пределов [3]. Приведем важное следствие этой теоремы, для чего нам потребуется ввести еще одно понятие.

Определение 5. Пусть $\{\|\cdot\|_n\}_{n=1}^\infty$ — определяющая система полунорм в пространстве Фреше E . Определим λ -топологию [1] в E^* как топологию индуктивного предела нормального разложения \overrightarrow{E}_ν^* :

$$E_\lambda^* = \varinjlim \overrightarrow{E}_\nu^* = \varinjlim_{n \rightarrow \infty} (E^{*n}, \|\cdot\|^n).$$

Сопряженное пространство $E_\lambda^{**} = (E_\lambda^*)^*$ к E_λ^* назовем вторым нормально сопряженным к E .

Заметим [2], что второе нормальное сопряженное E_λ^{**} связано с классическим вторым сопряженным E_β^{**} соотношением: $E_\beta^{**} \subset E_\lambda^{**}$ (при этом возможно $E_\beta^{**} \neq E_\lambda^{**}$). С помощью теоремы [3] можно доказать следующий результат.

Теорема 5. Функционал $\Lambda \in E_\lambda^{**}$ порождается некоторым элементом $x \in E$: $\Lambda f = \Lambda_x f = f(x)$ ($\forall f \in E^*$) тогда и только тогда, когда для некоторой базы окрестностей нуля $\mathcal{U} = \{U_\alpha\}_{\alpha \in A}$ в E все сужения $\Lambda|_{U_\alpha}^0$ (на поляры U_α^0 к U_α) *-слабо непрерывны.

Выражаю признательность научному руководителю И.В. Орлову за постановку задачи и внимание к работе.

ЛИТЕРАТУРА

- [1] Эдвардс Р. Функциональный анализ. Теория и приложения. // М.: Издательство "Мир", 1969. — 1071 с.
- [2] Orlov I. V. Normal functional indices and normal duality // Methods of Functional Analysis and Topology.-2002.-Vol.8.-№3—P.61-71.
- [3] I.V.Orlov Theorems on permutation of inductive and projective limit and application to generalized functions // Integral transforms and special functions.- 2009 (to appear).

Формулы Фон Неймана для кососимметрических аддитивных операторов

Джанапарова А. М.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА АЛГЕБРЫ И ФУНКЦИОНАЛЬНОГО АНАЛИЗА (ГРУППА 500)

В данной статье рассматриваются кососимметрические аддитивные операторы, действующие в гильбертовом пространстве. Для них определено понятие дефектного подпространства и приведены формулы фон Неймана для сопряженного оператора.

Работы М.М. Маламуда, В.А. Деркача (см. напр. [1], [2]) создали определенное направление в области исследования симметрических операторов в комплексном гильбертовом пространстве. Эти операторы охватывают такой важнейший класс линейных операторов, как дифференциальные операторы. В то же время для физических приложений представляют интерес кососимметрические аддитивные операторы. Поэтому возникает реальная проблема построить аналогичный аппарат для исследования таких операторов. В теории симметрических операторов центральное место занимает понятие дефектных подпространств, характеризующих отклонение оператора от самосопряженного. Поэтому, аналогично, рассмотрение кососимметрического оператора сводится к анализу его дефектных подпространств. Пусть \mathfrak{H} – комплексное гильбертово пространство.

Определение 6. *Аддитивный оператор $A : D_A \rightarrow R_A$ называется кососимметрическим, если $\overline{D_A} = \mathfrak{H}$ и $Re(Ax, y) = -Re(x, Ay)$, $\forall x, y \in D_A$.*

Из определения кососимметрического оператора вытекают следующие свойства:

- (1) Если A – кососимметрический аддитивный оператор, тогда $Re(Ax, x) = 0$, $\forall x \in D_A$.
- (2) Если A – кососимметрический аддитивный оператор, тогда $\sigma_p(A) \in i\mathbb{R}$. Собственные векторы x, y , соответствующие различным собственным значениям оператора A \mathbb{R} -ортогональны, т.е. $Re(x, y) = 0$. [3]
- (3) Если A – кососимметрический аддитивный оператор, тогда $\sigma_c(A) \in i\mathbb{R}$.

Определение 7. *Точка λ называется точкой регулярного типа оператора A , если $\exists k = k(\lambda) > 0$:*

$$\forall x \in D_A \quad \|(A - \lambda I)(x)\| \geq k\|x\|.$$

Множество всех точек регулярного типа оператора A называется полем регулярности этого оператора и обозначается $\pi(A)$.

Поле регулярности содержит резольвентное множество и является открытым.

Предложение 1. Пусть A – замкнутый оператор, действующий в гильбертовом пространстве \mathfrak{H} , и $\lambda \in \pi(A)$. Тогда \mathbb{R} -линеал $R_{A-\lambda I}$ замкнут и, таким образом является \mathbb{R} -подпространством пространства \mathfrak{H} .

Определение 8. Пусть A – кососимметрический аддитивный оператор, действующий в \mathfrak{H} . Подпространства вида

$$\mathfrak{N}_\lambda = \mathfrak{H}(-)R_A(\lambda),$$

где $R_A(\lambda) = R_{A-\lambda I}$, называются дефектными подпространствами оператора A .

Лемма 1. Пусть A – замкнутый кососимметрический оператор и $\lambda_0 \in \pi(A)$. Тогда в достаточно малой ε -окрестности точки λ_0 размерность дефектных подпространств \mathfrak{N}_λ оператора A не зависит от λ .

Лемма 2. В правой и левой полуплоскостях размерность дефектных подпространств \mathfrak{N}_λ замкнутого кососимметрического оператора A не зависит от λ .

Пусть A – замкнутый кососимметрический оператор и \mathfrak{N}_λ – дефектное подпространство. Рассмотрим числа:

$$n_+ = \dim \mathfrak{N}_1, \quad n_- = \dim \mathfrak{N}_{-1}. \quad (1)$$

На основании предыдущей теоремы

$$\dim \mathfrak{N}_\lambda = \begin{cases} n_+, & (Re \lambda > 0) \\ n_-, & (Re \lambda < 0) \end{cases}.$$

Числа n_+ и n_- , определяемые равенствами (1), называются дефектными числами оператора A , а их упорядоченную пара (n_+, n_-) – индексом дефекта оператора A .

Лемма 3. Если замкнутый кососимметрический оператор A имеет хотя бы одну мнимую точку регулярного типа, то его дефектные числа совпадают.

Предложение 2. Пусть $A : D_A \rightarrow R_A$ – кососимметрический аддитивный оператор в \mathfrak{H} и $\lambda \notin \sigma_p(A)$. Тогда \mathbb{R} -линеалы D_A и $\mathfrak{N}_{-\lambda}$ линейно независимы.

Следствие 1. При $\operatorname{Re} \lambda \neq 0$ мнeалы D_A и $\mathfrak{N}_{-\lambda}$ линейно независимы.

Теорема 6. Пусть A – замкнутый кососимметрический аддитивный оператор и $\operatorname{Re} \lambda \neq 0$. Тогда D_A , \mathfrak{N}_λ и $\mathfrak{N}_{-\bar{\lambda}}$ линейно независимы и

$$D_{A^{(*)}} = D_A \dot{+} \mathfrak{N}_\lambda \dot{+} \mathfrak{N}_{-\bar{\lambda}}; \quad (2)$$

$$\forall x \in D_{A^{(*)}} \quad x = x_0 + n_\lambda + n_{-\bar{\lambda}}, \quad x_0 \in D_A, \quad n_\lambda \in \mathfrak{N}_\lambda, \quad n_{-\bar{\lambda}} \in \mathfrak{N}_{-\bar{\lambda}} \quad (3)$$

$$A^{(*)}x = -Ax_0 + \bar{\lambda}n_\lambda - \lambda n_{-\bar{\lambda}}. \quad (4)$$

Формулы (2)-(4) будем называть обобщенными формулами фон Неймана.

Пример 2. Пусть $\mathfrak{H} = L_2[0, 1]$. Рассмотрим оператор, который определяется следующими равенствами:

$$D_A = \{x \in AC[0, 1] \mid x' \in L_2[0, 1], \operatorname{Re} x(0) = \operatorname{Re} x(1) = 0\},$$

$$\forall x \in D_A \quad (Ax)(t) = \operatorname{Re} x'(t).$$

Оператор A – кососимметрический.

Вычисления показывают, что дефектные подпространства оператора A имеют вид:

$$\mathfrak{N}_\lambda = \operatorname{span}_{\mathbb{R}} \left\{ e^{-\frac{|\lambda|^2}{\operatorname{Re} \lambda} t} \left(1 - \frac{i \operatorname{Im} \lambda}{\operatorname{Re} \lambda} \right) \right\},$$

где $\operatorname{span}_{\mathbb{R}}$ – вещественная линейная оболочка.

Тогда в силу обобщенных формул фон Неймана справедливы равенства:

$$D_A^{(*)} = D_A \dot{+} \mathfrak{N}_\lambda \dot{+} \mathfrak{N}_{-\bar{\lambda}};$$

$$\forall x \in D_{A^{(*)}} \quad x(t) = x_0(t) + n_\lambda(t) + n_{-\bar{\lambda}}(t);$$

$$(A^{(*)}x)(t) = -\operatorname{Re} x_0'(t) + \bar{\lambda}n_\lambda(t) - \lambda n_{-\bar{\lambda}}(t);$$

$$x_0(t) \in D_A, \quad n_\lambda(t) \in \mathfrak{N}_\lambda, \quad n_{-\bar{\lambda}}(t) \in \mathfrak{N}_{-\bar{\lambda}}.$$

Из вида дефектных подпространств также вытекает, что A – кососимметрический оператор с индексом дефекта $(1, 1)$.

ЛИТЕРАТУРА

- [1] S.Albeverio, J.Brasche, M.Malamud, H. Neidhardt. *Inverse Spectral Theory for Symmetric Operators with Several Gaps: Scalar-Type Weyl Functions* //University Bonn SFB 611, Preprint no 166, 2004.
- [2] V.Derkach. *On Weyl Function and Generalized Resolvents at a Hermitian operator in a Krein Space.* // Integral Equations and Operator Theory, Birkhauser Basel, Volume 23, Number 4, December 1995.

- [3] Третьяков Д.В. *Об одном отношении ортогональности в гильбертовом пространстве и некоторых свойствах аддитивных операторов.* // Ученые записки ТНУ, серия „Математика. Механика. Информатика и кибернетика.“ – 2004. – №2.

Компьютерная программа для развития внимания и восприятия детей с пониженным зрением

Донова Е. В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. Вернадского
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ ГРУППА 500-П

Целью работы является изучение особенностей внимания, восприятия и мышления слабовидящих, педагогических аспектов работы с такими детьми и как следствие, разработка комплекса коррекционных средств, в частности компьютерной программы с различными коррекционными упражнениями нескольких уровней сложности, с учётом гибких настроек монитора для каждого ребёнка в зависимости от ведущей офтальмопатологии.

В Украине по данным министерства охраны здоровья Украины на 2006 год насчитывалось более 10 000 слабовидящих детей. Помимо проблем со зрением, у многих из них наблюдаются различные психические и умственные отклонения. Такие дети нуждаются в специальных индивидуальных упражнениях для развития внимания, восприятия и мышления, требующих больших затрат времени со стороны педагога. На сегодняшний день не существует достаточного количества учителей способных уделить должное внимание каждому ребёнку. В этом случае, учитывая всё более развивающуюся компьютеризацию учебного процесса, было бы удобно воспользоваться персональным компьютером. Тогда функция педагога носила бы скорее организаторский, чем обучающий характер, за одним педагогом можно было бы закрепить целую группу детей. Этот подход не исключает индивидуальных занятий, и должен использоваться наряду с традиционным обучением.

Однако на сегодняшний день не существует программного обеспечения для детей с пониженным зрением для коррекции внимания и мышления.

Была проведена работа по определению различных проблем, возникающих у слабовидящих детей при работе за компьютером; консультации с офтальмологами, психологами, наблюдающими этих детей; исследовательская работа непосредственно с детьми.

В современных детских обучающих программах не предусмотрены варианты увеличения изображения, изменения контрастности и яркости, так необходимые для обеспечения нормального визуального восприятия информации детям с заболеваниями глазного анализатора. В разрабатываемой мною программе эти моменты учтены. Кроме того, для более лёгкого восприятия, меню

программы, а также все задания озвучены, при правильном или неправильном выполнении упражнений подаются звуковые сигналы. Так как программой пользуются дети, то интерфейс прост в управлении и понятен каждому ребёнку, все кнопки сопровождаются соответствующими надписями, картинки оформлены как можно ярче и красочнее.

В программе используются коррекционные и развивающие упражнения нескольких направлений. Прежде всего, это упражнения по развитию зрительного внимания. Детям предлагается из цветных либо чёрно-белых фигур разных форм и размеров найти одинаковые, выбрать фигуру, соответствующую заданному образцу, нажать как можно скорее на клавишу при появлении изображения в поле зрения (при этом фиксируется время реакции учащегося). Второе направление - формирование сенсорных эталонов по трём стратегиям: организация зрительного поиска ребёнком заданного элемента в системе эталонов по образцу, организация зрительного поиска эталона по представлению (мнемическому образу), формирование устойчивой связи между эталоном и словом. Задания основываются на формировании системы эталонов светлот, эталонов цвета, размера, ориентации, формы [1]. Детям нужно расположить фигуры различных цветов, размеров, форм, положений на плоскости, используя одну из вышеперечисленных стратегий. Третье направление - формирование перцептивных действий: развитие внимания ребёнка к форме и ориентации элементов мозаики; обучение выделению единого признака, по которому сгруппированы фигуры; обучение составлению из элементов мозаики простых изображений по образцу; обучение составлению по образцу нескольких изображений на одном поле; обучение опознанию неполного изображения и воссозданию его до целого; обучение восстановлению симметричного изображения до целого; обучение построению изображения по словесной инструкции; обучение выделению изображения в замаскированном перцептивном поле [1]. Четвёртое направление - формирование восприятия пространства и навыков ориентирования: развитие восприятия удалённости, тренировка глазных мышц в определении удалённости при установленной точке фиксации, обучение использованию светотеневых соотношений для восприятия объёма предмета, формирование умений точно оценивать расположение объектов, задачи-лабиринты, развитие умения использовать ориентиры в задачах поиска [1].

Отличительная особенность проекта в том, что специально для учителей разработан отдельный блок программы. В уже заготовленные шаблоны учитель может вносить новые картинки, определять правильные ответы, при этом

вся информация заносится в базу данных, откуда считывается при выполнении учеником того или иного упражнения. Таким образом, программа постоянно модифицируется, в ней появляются новые, интересные задания, уровень сложности которых задает учитель индивидуально для каждого ребёнка.

Реализация интерфейса, с которым работают дети, осуществляется в среде Macromedia Flash MX, с использованием анимации. Рисунки создаются с помощью ActionScript во Flash MX [2]. Учительский блок для создания заданий и упражнений разрабатывается в среде Eclipse [3] на языке Java [4].

Программа тестируется на базе учебно-реабилитационного центра для детей с нарушениями зрения г.Симферополя. Создана экспериментальная группа из 35 детей в возрасте от 7 до 10 лет с различной степенью зрительных и умственных нарушений. Каждому ребёнку предоставляется возможность пройти упражнения разных уровней сложности, начиная от элементарных, заканчивая довольно сложными логическими цепочками.

Помощь слабовидящим детям в развитии внимания, а также умения логически мыслить - вот основные задачи данной работы.

ЛИТЕРАТУРА

- [1] Григорьева Л. П., Бернадская М. Э. Блиникова И. В., Солнцева О. Г. *Развитие восприятия у ребенка. Пособие для коррекционных занятий с детьми с ослабленным зрением в семье, детском саду, начальной школе.* 2-е изд., дораб.- М.: Школьная пресса, 2007.- 72 с.: ил.
- [2] Д.Гурский. *Flash MX 2004 и Action Script 2.0: обучение на примерах.* Издательство: НОВОЕ ИЗДАНИЕ 2004 г.
- [3] Эрик Гамма, Кент Бек. *Eclipse.* Издательство: КУДИЦ-образ 2005.
- [4] Кей С. Хорстманн, Гари Корнелл. *Java 2. Библиотека профессионала. Том 1. Основы.* Издательство: Вильямс 2007 г.

Об одном условном G - ортонормированном базисе.

Дуплянкина Е.В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА АЛГЕБРЫ И ФУНКЦИОНАЛЬНОГО АНАЛИЗА (500)

ВВЕДЕНИЕ

В данной статье развиваются некоторые идеи, изложенные в работах В.А. Штрауса [2, 3], посвященных исследованию G - ортонормированных систем и базисов в гильбертовом пространстве.

ОБ ОДНОМ УСЛОВНОМ G - ОРТОНОРМИРОВАННОМ БАЗИСЕ.

Пусть \mathfrak{H} – гильбертово пространство. На \mathfrak{H} введена структура G - пространства при помощи ограниченного самосопряженного оператора G_α (см. [4]). Далее будем использовать определения базиса Шаудера, условного и безусловного базисов из функционального анализа.

В работе [1] указано, что система функций $\{|t|^\alpha e^{int}\}_{n \in \mathbb{Z}}$ является условным базисом пространства $L_2[-\pi, \pi]$ при $0 < |\alpha| < \frac{1}{2}$.

Пусть

$$0 > \alpha > -\frac{1}{2}. \quad (1)$$

Положим $\mathfrak{H} = L_2[-\pi, \pi]$. Зададим оператор $G_\alpha x(t) = \frac{1}{2\pi} |t|^{-2\alpha} x(t)$ и метрику $[x, y] = (G_\alpha x, y)$. G_α является ограниченным самосопряженным оператором. Действительно, в $L_2[-\pi, \pi]$ метрика задается таким образом: $[x, y] = \int_{-\pi}^{\pi} (G_\alpha x)(t) \overline{y(t)} dt$, где $t \in [-\pi, \pi]$, и исходя из этого можно показать симметричность и ограниченность оператора.

$$\begin{aligned} \text{Симметричность: } (G_\alpha x, y) &= \int_{-\pi}^{\pi} \frac{1}{2\pi} |t|^{-2\alpha} x(t) \overline{y(t)} dt = \int_{-\pi}^{\pi} \frac{1}{2\pi} t^{-2\alpha} x(t) \overline{y(t)} dt = \\ &= \int_{-\pi}^{\pi} x(t) \overline{\frac{1}{2\pi} t^{-2\alpha} y(t)} dt = (x, G_\alpha y); \end{aligned}$$

$$\begin{aligned} \text{Ограниченность: } \|G_\alpha x\|^2 &= \int_{-\pi}^{\pi} ((G_\alpha x)(t))^2 dt = \int_{-\pi}^{\pi} \left(\frac{1}{2\pi} |t|^{-2\alpha} x(t)\right)^2 dt = \\ &= \frac{1}{4\pi^2} \int_{-\pi}^{\pi} |t|^{-4\alpha} (x(t))^2 dt \leq \frac{1}{4\pi^2} \left(\max_{t \in [-\pi, \pi]} |t|^{-4\alpha}\right) \int_{-\pi}^{\pi} (x(t))^2 dt \stackrel{(1)}{=} \frac{1}{4} \|x\|^2. \end{aligned}$$

Рассмотрим систему функций $\{g_n\}_{n \in \mathbb{N}}$, где

$$g_1(t) = |t|^\alpha, \quad g_{2m}(t) = |t|^\alpha e^{-imt}, \quad g_{2m+1}(t) = |t|^\alpha e^{imt}, \quad m \in \mathbb{N}. \quad (2)$$

Покажем, что это условный G - ортонормированный базис в G - пространстве \mathfrak{N} . Нам известно, что система $\{f_k\}_{k \in \mathbb{Z}} = \{|t|^\alpha e^{ikt}\}_{k \in \mathbb{Z}}$ является условным базисом пространства $L_2[-\pi, \pi]$, а значит она является базисом Шаудера, т.е. для любого $x \in L_2[-\pi, \pi]$ и для любого $\varepsilon_1 > 0$ существует такой номер N_1 , начиная с которого верно неравенство

$$\left\| \sum_{k=-n}^n a_k f_k - x \right\| < \varepsilon_1 \quad (3)$$

и по определению найдутся такое $x \in L_2[-\pi, \pi]$ и перестановка в \mathbb{Z} $(s_k)_{k \in \mathbb{Z}}$, для которых существует такое $\varepsilon_2 > 0$, что начиная с некоторого номера N_2 выполняется неравенство

$$\left\| \sum_{k=-n}^n a_{s_k} f_{s_k} - x \right\| \geq \varepsilon_2. \quad (4)$$

Предложение 3. Пусть $\{f_k\}_{k \in \mathbb{Z}}$ - система векторов; $\{a_k\}_{k \in \mathbb{Z}}$ - последовательность комплексных чисел. Существует такая последовательность комплексных чисел $\{b_k\}_{k \in \mathbb{N}}$ и такая система векторов $\{g_k\}_{k \in \mathbb{N}}$, что для любой подстановки $(s_k)_{k \in \mathbb{Z}}$ ($s_k \in \mathbb{Z}$) существует такая подстановка $(\sigma_k)_{k \in \mathbb{N}}$, что выполняется равенство

$$\sum_{k=-n}^n a_{s_k} f_{s_k} = \sum_{p=1}^{2n+1} b_{\sigma_p} g_{\sigma_p}, \quad (5)$$

причем $(\sigma_k)_{k \in \mathbb{N}}$ - тождественная подстановка тогда и только тогда, когда $(s_k)_{k \in \mathbb{Z}}$ - тождественная.

Доказательство 1. Рассмотрим функцию

$$\varphi(k) := \begin{cases} \frac{k}{2}, & k - \text{четное}; \\ -\frac{k-1}{2}, & k - \text{нечетное}. \end{cases} \quad (6)$$

φ - биекция, при этом

$$\varphi(\overline{1, 2n+1}) = \overline{-n, n} \quad (7)$$

положим

$$\begin{aligned} g_k &:= f_{\varphi(k)} \quad (k \in \mathbb{N}); \\ b_k &:= a_{\varphi(k)} \quad (k \in \mathbb{N}). \end{aligned} \quad (8)$$

Пусть $(s_k)_{k \in \mathbb{Z}}$ - произвольная подстановка. Определим

$$\sigma_k := \varphi^{-1}(s_{\varphi(k)}) \quad (k \in \mathbb{N}) \quad (9)$$

Тогда

$$\sum_{k \in \overline{-n, n}} a_{s_k} f_{s_k} = \sum_{k \in \overline{-n, n}} a_{\varphi(\varphi^{-1}(s_k))} f_{\varphi(\varphi^{-1}(s_k))} \stackrel{(8)}{=} \sum_{k \in \overline{-n, n}} b_{\varphi^{-1}(s_k)} g_{\varphi^{-1}(s_k)}.$$

Обозначим $p = \varphi^{-1}(k)$. Когда k пробегает $\overline{-n, n}$, то в силу (7) p будет пробегать $\overline{1, 2n+1}$. Используя это вместе с равенством $\varphi^{-1}(s_k) \stackrel{(9)}{=} \varphi^{-1}(s_{\varphi(p)}) = \sigma_p$ получим

$$\sum_{k \in \overline{-n, n}} a_{s_k} f_{s_k} = \sum_{p \in \overline{1, 2n+1}} b_{\sigma_p} g_{\sigma_p}. \quad \blacksquare$$

Если $(s_k)_{k \in \mathbb{Z}}$ – тождественная подстановка, т.е. $s_k = k$, то по лемме подстановка $(\sigma_k)_{k \in \mathbb{N}}$ тоже тождественная и $\sum_{k=-n}^n a_k f_k = \sum_{p=1}^{2n+1} b_p g_p$. Подставив $\sum_{p=1}^{2n+1} b_p g_p$ в (3) получим, что для любого $x \in L_2[-\pi, \pi]$ и для любого $\varepsilon_1 > 0$ существует такой номер N_1 , начиная с которого верно неравенство

$$\left\| \sum_{k=1}^{2n+1} b_p g_p - x \right\| < \varepsilon_1,$$

следовательно $\{g_n\}_{n \in \mathbb{N}}$ – базис Шаудера.

Теперь, используя предложение 3, выберем по $(s_k)_{k \in \mathbb{Z}}$ из (4) подстановку $(\sigma_k)_{k \in \mathbb{N}}$ такую, что найдется $x \in L_2[-\pi, \pi]$, для которого существует $\varepsilon_2 > 0$, что начиная с некоторого номера N_2 выполняется неравенство

$$\left\| \sum_{p=1}^{2n+1} b_{\sigma_p} g_{\sigma_p} - x \right\| \geq \varepsilon_2,$$

значит $\{g_n\}_{n \in \mathbb{N}}$ – условный базис.

Прямым вычислением соответствующих интегралов нетрудно показать, что система функций (2) является G -ортонормированной, т.е. $||[g_k, g_j]|| = \delta_{kj}$. Например,

$$|[g_{2k}, g_{2m}]| = \left| \int_{-\pi}^{\pi} \frac{1}{2\pi} |t|^{-2\alpha} |t|^\alpha e^{-ikt} \overline{|t|^\alpha e^{-imt}} dt \right| = \left| \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{i(m-k)\pi} dt \right| = 0, \quad k \neq m \text{ и т.д.}$$

ЗАКЛЮЧЕНИЕ

В данной статье исследован пример G -ортонормированного условного базиса в G -пространстве $L_2[-\pi, \pi]$, где G -метрика задана при помощи весовой функции $\varphi_\alpha(t) = \frac{1}{2\pi} |t|^{-\alpha}$.

ЛИТЕРАТУРА

- [1] *Бабенко К.И.* О сопряженных функциях. ДАН СССР, т. 62, № 2, 1948 г., С. 154 – 160.
- [2] *Штраус В.А.* Оператор Грама и G -ортонормированные системы // Сб. аспирантов матем. фак-та, 1.Воронежск.ун-т., 1971 г., С 85 – 90.
- [3] *Штраус В.А.* G - ортонормированные системы и базисы в гильбертовом пространстве // Изв.вузов, математика. – 1973. –Вып.9 – С. 108 –117.
- [4] *Азизов Е.Я., Йохвидов И.С.* Основы теории линейных операторов в пространствах с индефинитной метрикой. – М.: Наука, 1986. – 352 с.

Построение медицинской экспертной системы логического типа.

Жолудева Т. В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. Вернадского
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-и)

В последнее время особенно активно ведутся разработки в области технологии экспертных систем, ведь эта технология является частью такого направления научных исследований, как искусственный интеллект. Своё практическое применение эти исследования нашли в тех областях жизни человека, где требуется обладание большим количеством знаний, определённой логикой мышления, опытом работы. Поэтому для создания такой системы требуются знания многих людей, экспертов в своей области. Разработка программного обеспечения, использующего знания экспертов, значительно повышает эффективность работы в тех сферах, в которых оно внедряется. В качестве примера типовых задач, решаемых экспертными системами, можно привести диагностику неисправностей в приборах, или даже человеческом организме; структурный анализ сложных объектов, таких как химические соединения; планирование последовательности выполнения работ, например, в промышленности.

Определение 9. [1] *Экспертная система - это система искусственного интеллекта, включающая знания об определенной слабо структурированной и трудноформализуемой узкой предметной области и способная предлагать и объяснять пользователю разумные решения. Экспертная система состоит из базы знаний, механизма логического вывода и подсистемы объяснений.*

База знаний рассматриваемой диагностирующей ЭС содержит записи, касающиеся знаний о конкретных диагнозах (гипотезах) и знаний о соответствующих симптомах (свидетельствах). Формат записи для описания конкретной гипотезы H (формат 1) может иметь следующий вид:

$$\text{НАЗВ. ГИП.}; P; S; (j_1, p_1^+, p_1^-); \dots; (j_s, p_s^+, p_s^-)$$

Здесь НАЗВ. ГИП. - название гипотезы H ; $P = P(H)$ - априорная (исходная) вероятность данной гипотезы; S - число свидетельств E_i , относящихся к данной гипотезе; j_k - номер свидетельства; $p_j^+ = P(E_{j_k}|H)$ - вероятность выполнения свидетельства для данной гипотезы; $p_j^- = P(E_{j_k}|\bar{H})$ - вероятность выполнения свидетельства j_k при неверности данной гипотезы .

Знания о свидетельствах могут быть представлены в следующем формате (формат 2):

Номер свидетельства; Задаваемый вопрос.

Рассмотрим фрагмент медицинской базы знаний.

- Описание гипотезы (формат 1):

Грипп; 0,001; 2; 1; 1; 0,01; 2; 0,9; 0,1

- Описание свидетельств (формат 2):

1; Есть ли у вас высокая температура?

2; Есть ли у вас насморк?

Дадим расшифровку приведенных записей. Априорная вероятность того, что некто болен гриппом, равна 0,001. В данной базе знаний с гриппом связаны два (2) симптома. Первый симптом - высокая температура. Вероятность высокой температуре здесь положена равной 1. Вероятность высокой температуры при отсутствии гриппа равна 0,01.

Второй симптом - насморк. Вероятность насморка при гриппе равна 0,9. Вероятность насморка при отсутствии гриппа равна 0,1.

На симптом № 1 (температура) могут быть ссылки и из записей для других гипотез (болезней), но вероятности p^+ , p^- - в тройках (l, p^+, p^-) будут уже другими.

В алгоритме логического вывода используется цифровое шкалирование ответов. Но пользователь программы не всегда сможет оценить своё состояние объективно чётко в соответствии с этой шкалой. Поэтому для удобства вопросы разбиты на классы, в зависимости от вариантов ответа на них. Пользователю предоставляется возможность видеть удобные для восприятия варианты оценивания его состояния, не вызывающие сомнений и потребности в дополнительных уточнениях при работе с программой. Каждому классу ставится в соответствие отдельный диапазон цифровых значений ответов. При пересчёте априорных вероятностей гипотез в качестве коэффициента мы берём отношение цифрового значения ответа к количеству ответов для данного класса свидетельств.

Такое усовершенствование алгоритма Нейлора позволяет получить более чёткие и правильные ответы, а, следовательно, позволит алгоритму работать эффективнее.

ЛІТЕРАТУРА

- [1] Нейлор К. Как построить свою экспертную систему: Пер. с англ. — М.: Энергоатомиздат, 1991. - 286 с.
- [2] Черноруцкий И. Методы принятия решений, БХВ-Петербург.— 2005. - 416 с.
- [3] Гаврилов А.В., Новицкая Ю.В. Разработка экспертных систем, БХВ-Петербург.— 2006. - 124 с.

Вычисление BWT за линейное время

Заднипряный М.В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (500и)

ВВЕДЕНИЕ

В настоящей статье предлагается реализация BWT с использованием суффиксных деревьев. Текст, после преобразования Барроуза-Уилера, с помощью словарного метода, сжимается гораздо лучше, однако прямое решение данной задачи требует квадратичных затрат памяти, что на практике делает его неприменимым, также оно требует применения алгоритма сортировки. Реализация BWT с помощью метода Патриция, даёт приемлемые затраты памяти, но громоздкость самого метода, делает его малоприменимым на практике. Отличительной особенностью предлагаемого ниже алгоритма является его линейность и относительная простота реализации.

BWT ЗА ЛИНЕЙНОЕ ВРЕМЯ: АЛГОРИТМ

Суффиксное дерево $S(T)$ – это способ представления текста T . Чтобы построить $S(T)$ для текста $T = t_1 \dots t_n$, нужно приписать специальный символ $\$$ в конец текста, взять все $n + 1$ суффиксов, подвесить их за начала и склеить все ветки, идущие по одинаковым буквам. В каждом листе записывается номер суффикса, заканчивающегося в этом листе. Номером суффикса является индекс его начала в тексте T .

Заметим, что ни один суффикс в $S(T)$ не может полностью лежать в другом суффиксе, поскольку они заканчиваются специальным символом $\$$. Таким образом, листьев в $S(T)$ всегда будет $n + 1$ для строки $t_1 \dots t_n$, то есть столько же, сколько суффиксов.

Дописав в конец текста $\$$, построим суффиксное дерево. Последовательно рассмотрим все внутренние вершины и отсортируем детей каждой из них в алфавитном порядке. Вспомним, что в каждом листе хранится номер конкретного суффикса. Двигаясь сверху вниз, составим последовательность из чисел во всех листьях. Теперь просматриваем эти числа и для каждого числа i пишем букву s_{i-1} , полученная последовательность букв будет результатом преобразования.

Обход всех листьев за линейное время можно осуществить, используя обход в глубину.

Убедимся, что полученная последовательность и есть последний столбец матрицы сдвигов, в которой строчки отсортированы в алфавитном порядке.

Заметим, что алфавитный порядок суффиксов, совпадает, с алфавитным порядком сдвигов.

Рассмотрим некоторый текст, допишем \$, сравним два произвольных суффикса: до какого-то момента суффиксы могут быть одинаковы.

Найдём место, где появляется расхождение. Рассмотрим отдельно случай, когда это будут различные буквы, например b и c , тогда суффикс, содержащий b , будет идти раньше идти раньше в алфавитном порядке и случай, когда вместо одной из букв будет символ \$, так как \$ ввели как последний символ алфавита, то он окажется на нижнем ребре.

Если дописать к суффиксам после символа \$ оставшиеся до циклического сдвига буквы, то алфавитный порядок сохранится.

Рассмотрим ещё одно полезное наблюдение. В листьях мы читаем порядок стартов суффиксов. Пусть i - это позиция, с которой начинается соответствующий суффикс, а следовательно и сдвиг. Тогда s_{i-1} - это последняя буква сдвига. Прочитаем все буквы s_{i-1} в такой очерёдности, что последовательность букв s_i образует алфавитный порядок. Тогда последовательность последних букв сдвигов s_{i-1} будет результат BWT.

ЗАКЛЮЧЕНИЕ

В статье рассмотрен алгоритм построения BWT за линейное время с помощью суффиксных деревьев.

ЛИТЕРАТУРА

- [1] *M. Burrows and D. Wheeler*. A block sorting losless data compression algorithm
<http://www.hp1.hp.com/techreports/Compaq-DEC/SRC-RR-124.pdf>
- [2] *Mark Nelson*. Data Compression with the Burrows-Wheeler Transform
<http://marknelson.ua/1996/09/01/bwt>
- [3] *Esko Ukkonen*. Suffix tree and suffix array techniques for pattern analysis in strings
<http://www.cs.helsinki.fi/u/ukkonen/Erice2005.ppt>

Функции, имеющие отношение к дзета-функции Римана

Зайцева А. Ю.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ (ГРУППА 500-П)

Целью работы является изучение нового класса функций, имеющих отношение к дзета-функции Римана, но обладающими собственными свойствами. Дзета-функция Римана $\zeta(s)$ определена с помощью ряда Дирихле:

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s} \quad (1)$$

В области $\{s: \operatorname{Re}(s) > 1\}$, этот ряд сходится, является аналитической функцией и допускает аналитическое продолжение на всю комплексную плоскость без единицы. В этой области также верно представление в виде бесконечного произведения (тождество Эйлера)

$$\zeta(s) = \prod_p \frac{1}{1-p^{-s}}, \quad (2)$$

где произведение берётся по всем простым числам p . Это равенство представляет собой одно из основных свойств дзета-функции. [1] Определим функции, имеющие отношение к функции Римана.

$$S(\alpha, \lambda) = \sum_{n=2}^{\infty} \frac{\sin(\lambda \cdot n^\alpha)}{n^\alpha} \quad (3)$$

$$C(\alpha, \lambda) = \sum_{n=2}^{\infty} \frac{\cos(\lambda \cdot n^\alpha)}{n^\alpha} \quad (4)$$

Рассмотрим поведение функций (3) и (4) при ограниченном n . При параметрах $\lambda = 1$ и $\alpha = 1$ данные конечные суммы являются рядами Фурье и соответственно обладают его свойствами. Например при изучении усеченного ряда Фурье возникает случай, имеющий название явление Гиббса. Явлением Гиббса называется особенность поведения частичных сумм ряда Фурье в окрестности точки разрыва функции, впервые обнаруженная Уилбрейамом в 1848 г. и позже переоткрытая Гиббсом в 1898 г. Явление Гиббса можно ожидать всякий раз, когда функция имеет разрыв. [2]

Его нельзя преодолеть, но можно сгладить путем применения фильтров. Существуют различные их виды, как например, сглаживание Ланцоша, сглаживание Фейера, окно Хемминга, окно фон Ганна. К функциям (3) и (4) были применены все вышеперечисленные приемы, однако наилучшей для данных

функций показала себя последовательность, получившая название весовой функции Хэмминга. [2]

$$d_k = \begin{cases} \{0.46 + 0.54 * \cos(\pi * k) / N, |k| \leq N \\ 0, |k| > N \end{cases} \quad (5)$$

Так же было рассмотрено построение графиков функций с помощью асимптотического приближения. Согласно [3], функции (3) и (4) при $\alpha = 1$ могут быть заменены на функции вида (6) и (7) соответственно:

$$S := \frac{\pi - \lambda}{2} \quad (6)$$

$$C := 0.5 \left(\ln \frac{1}{2(1 - \cos \lambda)} \right). \quad (7)$$

В общем случае ряды для любого λ при $\alpha > 1$ сходятся, так как

$$|S(\alpha, \lambda)| \leq \sum_{n=2}^{\infty} \frac{1}{n^\alpha} = \zeta(\alpha) - 1 < \infty \quad (8)$$

$$|C(\alpha, \lambda)| \leq \sum_{n=2}^{\infty} \frac{1}{n^\alpha} = \zeta(\alpha) - 1 < \infty \quad (9)$$

В результате дальнейшего изучения были получены кривые, как изображения функции $L(C(\alpha, \lambda), S(\alpha, \lambda))$, где эти функции выступают в качестве параметров. При $0 < \alpha < 1$ кривая имеет форму спирали, и минимальное изменение параметра влечет за собой изменение её формы. При $\alpha = 2$ кривая имеет форму листа, причем она обладает фрактальной формой. При более детальном изучении этого “листа” Римана выяснилось, что кривая периодична с периодом 2π .

ЛИТЕРАТУРА

- [1] Воронин С.М., Карацуба А.А. *Дзета-функция Римана*. – М.: Физматлит, 1994. – 376 с.
- [2] Хемминг Р.В. *Цифровые фильтры*. – М.: Сов. радио, 1980. – 224 с., ил.
- [3] Градштейн И.С., Рыжик И.М. *Таблицы интегралов, сумм, рядов и производений*. – М.: Физматгиз, 1963. – 1100 с., ил.

Малые колебания системы идеальных стратифицированных жидкостей

Имрякова М.А.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА МАТЕМАТИЧЕСКОГО АНАЛИЗА (ГРУППА 500)

В данной работе исследуются малые (линейные) движения системы идеальных стратифицированных жидкостей, находящихся в цилиндрическом сосуде с произвольным поперечным сечением. Изучаются собственные колебания гидросистемы. Спектральная задача решается с помощью метода Фурье разделения переменных. В результате получаются два трансцендентных уравнения относительно неизвестных частот колебаний, графическое исследование которых показывает, что рассматриваемая задача имеет в качестве решений два типа волн: поверхностные и внутренние.

Будем считать, что идеальные стратифицированные жидкости, плотности которых в состоянии покоя изменяются вдоль вертикальной оси, полностью заполняют сосуд произвольной формы и занимают связанные области Ω_k ($k = \overline{1, n}$), ограниченные твердыми стенками S_k ($k = \overline{1, n}$) и границами раздела жидкостей Γ_k ($k = \overline{1, n-1}$).

Введем систему координат $Ox_1x_2x_3$, жестко связанную с сосудом так, что ось Ox_3 направлена против действия силы тяжести: $\vec{g} = -g\vec{e}_3$.

Малые движения исходной гидросистемы описываются следующей эволюционной задачей:

$$\begin{cases} \rho_{0k}(x_3) \frac{\partial \vec{u}_k}{\partial t} = -\nabla p_k - g\rho_k \vec{e}_3 & (\text{в } \Omega_k, k = \overline{1, n}), \\ \frac{\partial \rho_k}{\partial t} + \nabla \rho_{0k} \frac{\partial \vec{u}_k}{\partial t} = 0, \quad \nabla \cdot \vec{u}_k = 0 & (\text{в } \Omega_k, k = \overline{1, n}), \end{cases} \quad (1)$$

$$\vec{u}_k \cdot \vec{n}_k = 0 \quad (\text{на } S_k, k = \overline{1, n}), \quad (2)$$

$$p_k - p_{k+1} = g(\Delta \rho_{0k}) \zeta_k \quad (\text{на } \Gamma_k, k = \overline{1, n-1}), \quad (3)$$

$$\zeta_k = (u_3)_k = (u_3)_{k+1} \quad (\text{на } \Gamma_k, k = \overline{1, n-1}), \quad (4)$$

с начальными условиями:

$$\begin{cases} \zeta_k(0, x_1, x_2) = \zeta_k^0(x_1, x_2) & (k = \overline{1, n-1}), \\ \rho_k(0, x) = \rho_k^0(x) & (x \in \Omega_k, k = \overline{1, n}), \\ \vec{u}_k(0, x) = \vec{u}_k^0 & (k = \overline{1, n}), \end{cases} \quad (5)$$

где $\vec{u}_k(t, x)$ — поле скоростей в жидкости, $p_k(t, x)$ и $\rho_k(t, x)$ — отклонения поля давлений и плотности от равновесных $p_{0k}(x_3)$ и $\rho_{0k}(x_3)$ соответственно, $\Delta\rho_{0k}(x_3) = \rho_{0k} - \rho_{0,k+1}$ — скачок плотности на границах раздела Γ_k , ζ_k — вертикальное отклонение движущейся поверхности $\hat{\Gamma}_k(t)$ от равновесной горизонтальной поверхности Γ_k , \vec{n}_k — нормаль к твердой стенке S_k .

Далее будем изучать лишь собственные колебания исходной гидросистемы, т.е. такие решения (1)–(5), для которых временная переменная отделяется, причем $\vec{u}_k(t, x)$, $p_k(t, x)$, $\rho_k(t, x)$, $\zeta_k(t, \hat{x})$ ($\hat{x} \in \Gamma_k$) зависят от времени по закону $\exp(i\omega t)$, где ω — неизвестная частота колебаний. Тогда задача (1)–(4) приводит к следующей спектральной задаче:

$$\begin{cases} \omega^2 \rho_{0k}(x_3) \vec{u}_k = \nabla p_k + g \rho_k \vec{e}_3 & (\text{в } \Omega_k, \quad k = \overline{1, n}), \\ \rho_k + \rho'_{0k}(x_3)(u_3)_k = 0, \quad \nabla \vec{u}_k = 0 & (\text{в } \Omega_k, \quad k = \overline{1, n}). \end{cases} \quad (6)$$

с условиями (2)–(4).

Рассмотрим частный случай данной задачи для сосуда цилиндрической формы с произвольным поперечным сечением Γ и будем считать $N_k^2(x_3) = N_k^2 \Rightarrow \rho_{0k}(x_3) = \rho_{0k}(0) \exp(-\varepsilon_k x_3)$, где $\varepsilon_k = \frac{N_k^2}{g}$, $N_k^2(x_3) := -g\rho'_{0k}(x_3)/\rho_{0k}(x_3)$ — частота плавучести k -ой жидкости. Исключая все переменные, кроме вертикального смещения $w_k = (u_3)_k$, получаем задачу:

$$(N_k^2 - \omega^2) \Delta_2 w_k - \omega^2 \left(\frac{\partial^2 w_k}{\partial x_3^2} - \frac{1}{g} N_k^2 \frac{\partial w_k}{\partial x_3} \right) = 0 \quad (k = \overline{1, n}), \quad (7)$$

$$w_1 = 0, \quad w_n = 0, \quad (8)$$

$$\frac{\partial w_k}{\partial \vec{n}_0} = 0 \quad (k = \overline{1, n}), \quad (9)$$

$$\rho_{0k} \left(\omega^2 \frac{\partial w_k}{\partial x_3} + g \Delta_2 w_k \right) = \rho_{0,k+1} \left(\omega^2 \frac{\partial w_{k+1}}{\partial x_3} + g \Delta_2 w_{k+1} \right) \quad (k = \overline{1, n-1}), \quad (10)$$

$$\zeta_k = w_k = w_{k+1} \quad (k = \overline{1, n-1}). \quad (11)$$

(8) — это граничное условие на нижнем и верхнем основаниях цилиндрического сосуда, \vec{n}_0 — нормаль к боковой стенке S .

Решим задачу (7)–(11) методом разделения переменных, представив $w_k = Z_k(x_3) \cdot X(x_1, x_2)$. Получим две задачи:

$$\Delta_2 X(x_1, x_2) + \mu^2 X(x_1, x_2) = 0, \quad \frac{\partial X}{\partial n_0} \Big|_{\partial \Gamma} = 0, \quad \int_{\Gamma} X dx_1 dx_2 = 0, \quad (12)$$

$$\frac{1}{\mu^2} (\rho_{0k} Z_k')' - \rho_{0k} Z_k = \lambda g \rho'_{0k} Z_k \quad (k = \overline{1, n}), \quad (13)$$

$$Z_1(0) = Z_n(h) = 0, \quad (14)$$

$$Z_k(h_k) = Z_{k+1}(h_k) \quad (k = \overline{1, n-1}), \quad (15)$$

$$\rho_{0k} Z'_k(h_k) - \rho_{0, k+1} Z'_{k+1}(h_k) = \lambda \mu^2 g (\rho_{0k} - \rho_{0, k+1}) Z_k(h_k) \quad (k = \overline{1, n-1}). \quad (16)$$

В задаче (13)–(16) $\lambda = \omega^{-2}$.

Задача (12) имеет дискретный спектр $\{\mu_p\}_{p=1}^{\infty}$: $0 < \mu_1 \leq \mu_2 \leq \dots \leq \mu_p \leq \dots, \mu_p \rightarrow +\infty (p \rightarrow \infty)$.

Решим задачу (13)–(16) для двух жидкостей, считая, что $\Omega_1 = \Gamma \times [0, h_1]$, $\Omega_2 = \Gamma \times [h_1, h - h_1]$, где h — высота сосуда. Получим два трансцендентных уравнения относительно $\lambda = \omega^{-2}$:

$$\begin{aligned} 1) \quad & \lambda \mu_p^2 g (\rho_{01} - \rho_{02}) - \frac{1}{2} (\rho_{01} \varepsilon_1 - \rho_{02} \varepsilon_2) = \\ & = \rho_{01} q_1 \operatorname{ctg}(q_1 h_1) + \rho_{02} q_2 \operatorname{ctg}(q_2 (h - h_1)) \end{aligned} \quad (17)$$

$$\text{при } \lambda > \frac{1}{N_k^2} + \frac{N_k^2}{4g^2 \mu_p^2}, \quad -q_k^2 = \frac{1}{4} \varepsilon_k^2 - \mu_p^2 (N_k^2 \lambda - 1) < 0;$$

$$\begin{aligned} 2) \quad & \lambda \mu_p^2 g (\rho_{01} - \rho_{02}) - \left(\rho_{01} \frac{\varepsilon_1}{2} - \rho_{02} \varepsilon_2 \right) = \\ & = \rho_{01} p_1 \operatorname{cth}(p_1 h_1) + 2\rho_{02} p_2 \operatorname{cth}(p_2 (h - h_1)) \end{aligned} \quad (18)$$

$$\text{при } \lambda < \frac{1}{N_k^2} + \frac{N_k^2}{4g^2 \mu_p^2}, \quad p^2 = \frac{1}{4} \varepsilon_k^2 - \mu_p^2 (N_k^2 \lambda - 1) > 0.$$

Графическое исследование (17) и (18) привело к следующим выводам.

Случаю (17) для любого p соответствует счетное множество собственных значений $\lambda_{pk} \rightarrow +\infty (k \rightarrow \infty)$. Следовательно, $\omega_{pk} \rightarrow 0 (p \rightarrow \infty, k \rightarrow \infty)$.

Случаю (18) для любого p соответствует одно собственное значение λ_p . Следовательно, $\omega_p \rightarrow +\infty (p \rightarrow \infty)$.

Таким образом, рассматриваемая задача имеет в качестве решений поверхностные волны, родственные поверхностным колебаниям однородной нестратифицированной жидкости, а также внутренние волны, обусловленные стратификацией жидкостей.

ЛІТЕРАТУРА

- [1] Ландау Л.Д., Лифшиц Е.М. *Гидродинамика* // М.: "Наука", — 1986.
- [2] Мышкис А.Д., Бабский В.Г., Жуков М.Е., Копачевский Н.Д. и др. *Методы решения задач гидромеханики для условий невесомости* // Киев: Наук. думка, — 1992.

Преимущества и недостатки систем управления распределенными базами данных

Кемельман М. Р.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. Вернадского
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

Распределенные системы призваны разрешить проблему островов информации. Базы данных иногда рассматривают как некие электронные острова, представляющие собой отдельные и, в общем случае, труднодоступные места, подобные удаленным друг от друга островам. Данное положение может являться следствием географической разобщенности, несовместимости используемой компьютерной архитектуры, несовместимости используемых коммутационных протоколов и т.д. Интеграция отдельных баз данных в одно логическое целое способна изменить подобное положение дел.

Определение 10. *Распределенная база данных - это набор логически связанных между собой разделяемых данных (и их описаний), которые физически распределены в некоторой компьютерной сети. Тогда распределенная СУБД - это программный комплекс, предназначенный для управления распределенными базами данных и позволяющий сделать распределенность информации прозрачной для конечного пользователя [1].*

Система управления распределенными базами данных (СУРБД) состоит из единой логической базы данных, разделенной на некоторое количество фрагментов. Каждый фрагмент базы данных сохраняется на одном или нескольких компьютерах, которые соединены между собой линиями связи и каждый из которых работает под управлением отдельной СУБД. Любой из сайтов способен независимо обрабатывать запросы пользователей, требующие доступа к локально сохраняемым данным (что создает определенную степень локальной автономии), а также способен обрабатывать данные, сохраняемые на других компьютерах сети [2].

Системы с распределенными базами данных имеют преимущества перед традиционными централизованными системами баз данных:

1. Отражение структуры организации. Крупные организации, как правило, имеют множество отделений, которые могут находиться в разных концах страны и даже за ее пределами. Вполне логично будет предположить, что используемая компанией база данных должна быть распределена между ее отдельными офисами. В подобной базе данных персонал отделения сможет

выполнять необходимые ему локальные запросы. Руководству компании может потребоваться выполнять глобальные запросы, предусматривающие получение доступа к данным, сохраняемым во всех существующих отделениях компании. Географическая распределенность организации может быть отражена в распределении ее данных, причем пользователи одного сайта смогут получать доступ к данным, сохраняемым на других сайтах. Данные могут быть помещены на тот сайт, на котором зарегистрированы пользователи, которые их чаще всего используют.

2. Повышение доступности данных. В централизованных СУБД отказ центрального компьютера вызывает прекращение функционирования всей СУБД. Однако отказ одного из сайтов СУРБД или линии связи между сайтами делает недоступными лишь некоторые сайты, тогда как вся система в целом сохранит свою работоспособность.

3. Повышение надежности. Если организована репликация данных, в результате чего данные и их копии будут размещены на более чем одном сайте, отказ отдельного узла или соединительной связи между узлами не приведет к недоступности данных в системе.

4. Повышение производительности. Если данные размещены на самом нагруженном сайте, который унаследовал от систем-предшественников высокий уровень параллельности обработки, то развертывание распределенной СУБД может способствовать повышению скорости доступа к базе данных (по сравнению с доступом к удаленной централизованной СУБД). Более того, поскольку каждый сайт работает только с частью базы данных, уровень использования центрального процессора и служб ввода/вывода может оказаться ниже, чем в случае централизованной СУБД.

5. Экономические выгоды. В шестидесятые годы мощность вычислительной установки возрастала пропорционально квадрату стоимости ее оборудования, поэтому система, стоимость которой была втрое выше стоимости данной, превосходила ее по мощности в девять раз. Эта зависимость получила название закона Гроша (Grosch). Однако в настоящее время считается общепринятым положение, согласно которому намного дешевле собрать из небольших компьютеров систему, мощность которой будет эквивалентна мощности одного большого компьютера [3].

К сожалению, системы с распределенными базами данных не лишены некоторых недостатков:

1. Повышение сложности. Распределенные СУБД, способные скрыть от конечных пользователей распределенную природу используемых ими данных

и обеспечить необходимый уровень производительности, надежности и доступности, безусловно являются более сложными программными комплексами, чем централизованные СУБД.

2. Увеличение стоимости. Увеличение сложности означает и увеличение затрат на приобретение и сопровождение СУРБД (по сравнению с обычными централизованными СУБД). Разворачивание распределенной СУБД потребует дополнительного оборудования, необходимого для установки сетевых соединений между сайтами. Следует ожидать и роста расходов на оплату каналов связи, вызванных возрастанием сетевого графика.

3. Проблемы защиты. В централизованных системах доступ к данным легко контролируется. Однако в распределенных системах потребуется организовать контроль доступа не только к данным, реплицируемым на несколько различных сайтов, но и защиту сетевых соединений самих по себе.

4. Отсутствие стандартов. Хотя вполне очевидно, что функционирование распределенных СУБД зависит от эффективности используемых каналов связи, только в последнее время стали вырисовываться контуры стандарта на каналы связи и протоколы доступа к данным. Отсутствие стандартов существенно ограничивает потенциальные возможности распределенных СУБД. Кроме того, не существует инструментальных средств и методологий, способных помочь пользователям в преобразовании централизованных систем в распределенные.

5. Недостаток опыта. В настоящее время в эксплуатации находится уже несколько систем-прототипов и распределенных СУБД специального назначения, что позволило уточнить требования к используемым протоколам и установить круг основных проблем. Однако на текущий момент распределенные системы общего назначения еще не получили широкого распространения. Соответственно еще не накоплен необходимый опыт промышленной эксплуатации распределенных систем, сравнимый с опытом эксплуатации централизованных систем.

ЛИТЕРАТУРА

- [1] Михайлов М. *Распределенные базы данных*. КомпьютерПресс. 1991. - N 4. - С.70-74
- [2] Кокарев В.Н. *Распределенные СУБД. Сети*. - 1993. -N 3. - С.20-28
- [3] Ривкин М.Н. *Распределенные СУБД*. Мир ПК.- 1993. -N 5. - С.40-46

Алгоритм вычисления размерности и роста алгебр, заданных образующими и соотношениями

Канищев С.А.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

Задача исследования размерности и роста алгебр, заданных образующими и соотношениями относится к классическим задачам общей алгебры [1]. Аналитические средства и методы решения при этом зачастую дополняются алгоритмами компьютерной алгебры [2]. В настоящей работе реализация алгоритма Бухбергера построения базиса Грёбнера позволяет сформулировать и доказать теорему о виде базиса Грёбнера алгебры, заданной образующими и соотношениями групповой алгебры группы Кокстера A_n и соотношениями ортогональности и как следствие получить размерность, исследуемой алгебры.

Обозначим через $\mathfrak{A} = \mathbb{C} \langle X \rangle$ свободную ассоциативную алгебру с единицей, где $X = \{x_1, x_2, \dots, x_n\}$ – это множество букв, которые принято называть образующими алгебры. Множество слов, включая пустое слово Λ , обозначим через S . Пусть I – идеал, порождённый соотношениями $f_1(x_1, x_2, \dots, x_n) = 0, f_2(x_1, x_2, \dots, x_n) = 0, \dots, f_r(x_1, x_2, \dots, x_n) = 0$.

Определение 11. Алгеброй A , заданной образующими x_1, x_2, \dots, x_n и соотношениями $f_1 = 0, f_2 = 0, \dots, f_r = 0$ называется факторалгебра $\mathfrak{A}|_I$ (обозначается $A = \mathbb{C} \langle x_1, x_2, \dots, x_n | f_1 = 0, f_2 = 0, \dots, f_r = 0 \rangle$).

Слово $s \in S$ называется нормальным (по модулю идеала I), если s не является старшим словом никакого элемента I . Обозначим через N линейную оболочку множества нормальных слов, которую мы назовем нормальным дополнением идеала I . Имеет место разложение в прямую сумму векторных пространств: $\mathfrak{A} = N \oplus I$ [2]. Отсюда следует, что алгебра A и соответствующая ей фактор-алгебра $\mathfrak{A}|_I$ изоморфны алгебре N со специальной операцией умножения, откуда вытекает, что количество нормальных слов равно $\dim A = \dim N$.

Следовательно, если мы научимся находить нормальные слова и считать их количество, то тем самым мы будем иметь способ вычисления размерности и роста алгебры $\mathfrak{A}|_I$. Оказывается [2], можно считать нормальные слова не с помощью всего идеала I , а с помощью некоторого его подмножества. Подмножество G идеала I называется его базисом Грёбнера, если для $\forall v \in I \exists g \in G$ такое, что старшее слово g является подсловом старшего слова v . При этом

слово s нормально тогда и только тогда, когда ни одно старшее слово $\forall g \in G$ не является подсловом слова s .

Построив базис Грёбнера можно посчитать нормальные слова и узнать размерность и рост алгебры.

Рассмотрим следующие алгебры, заданные образующими и соотношениями:

$$A_{n,\tau} = \mathbb{C}\langle p_1, p_2, \dots, p_n \mid \begin{array}{ll} p_i^2 = p_i & i = \overline{1, n}; \\ p_i p_j p_i - \tau p_i = p_j p_i p_j - \tau p_j & |i - j| = 1; \\ p_i p_j = 0 & |i - j| > 1 \end{array} \rangle$$

$$A_{n,\tau,\perp} = \mathbb{C}\langle p_1, p_2, \dots, p_n \mid \begin{array}{ll} p_i^2 = p_i & i = \overline{1, n}; \\ p_i p_j p_i - \tau p_i = 0 & |i - j| = 1; \\ p_i p_j = 0 & |i - j| > 1 \end{array} \rangle$$

В статье [3] доказывается, что $\dim A_{n,\tau,\perp} = n^2 + 1$

Реализация алгоритма Бухбергера построения базиса Грёбнера позволяет получить следующую теорему:

Теорема 7. *Множество старших слов базиса Грёбнера алгебры $A_{n,\tau}$ совпадает со множеством старших слов базиса Грёбнера алгебры $A_{n,\tau,\perp}$.*

Следствие 2. $\dim A_{n,\tau} = \dim A_{n,\tau,\perp} = n^2 + 1$.

ЛИТЕРАТУРА

- [1] Ostrovskiy V. L., Samoilenko Yu. S. *Introduction to the theory representation of finitely presented *-algebras. 1. Representations by bounded operators. Rev. Math. & Math. Phys., Gordon and Breach.* – 1999. – Vol.11. – P.1-261.
- [2] Уфнаровский В. А. *ИНТ. Современные проблемы математики. Фундаментальные направления. Т. 57. I. Комбинаторные и асимптотические методы в алгебре.* – М.: ВИНТИ, 1990. – 272 с.
- [3] Власенка М. А., Попова Н. Д. *О конфигурациях подпространств гильбертова пространства с фиксированными углами между ними // Укр. Мат. Журн.* – 56, №5 – 2004. – С. 606-615.

Методы анализа затрат в программном обеспечении задач управления предприятием

Карпенко Е.А.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. Вернадского
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

При разработке программного обеспечения задач управления необходим учет распределения затрат, расчета себестоимости, повышения прибыльности предприятия.

Существуют четыре наиболее используемые классификационные категории затрат:

1. *По отношению к изменениям в деятельности.* К ним относятся постоянные затраты, которые не изменяются пропорционально увеличению объемов деятельности компании, например такие, как амортизация или оплата труда руководства

2. *По непосредственному отношению к продукции или к другим объектам затрат.* Сюда входят косвенные затраты, такие, как зарплата торговых представителей или расходы на энергообеспечение вспомогательных производственных мощностей, отношение которых к конкретной продукции должно определяться с помощью дальнейшего анализа.

3. *По восприимчивости к управленческому влиянию на определенном уровне ответственности.* Это регулируемые затраты, на которые в ходе обсуждения может оказывать воздействие определенное управленческое подразделение, в сравнении с нерегулируемыми затратами, на которые повлиять нельзя.

4. *По значимости для оценки запаса.*

Это производственные затраты - такие, как труд производственных рабочих, сырье и вспомогательные материалы, планирование и контроль производства, которые прямо относятся на счет запасов.

В связи с тем, что вопрос выбора метода расчета себестоимости и распределения затрат может оказаться одним из ключевых факторов правильности управленческих решений определим понятие „себестоимость“. Себестоимость какого-либо объекта означает величину затрат предприятия на производство данного объекта. Определим понятие „метод расчета себестоимости объекта“ как способ отнесения или распределения части затрат предприятия на данный объект.

В простейшей системе планирования и контроля устанавливаются ожидаемые уровни затрат на каждый период по каждой стоимостной категории (труд, сырье, энергия, зарплата административного персонала и т.д.) и также

по центрам учета ответственности. На конец каждого периода в данной системе расчета сравниваются плановые с фактическими затратами и отмечаются расхождения.

Таким образом, подобная простейшая система постоянно фиксирует расхождения, вызываемые исключительно изменениями объема, и поэтому данная информация представляет небольшой интерес для управления. Исходя из выше сказанного, необходима такая система, которая автоматически приспосабливается к изменениям в объемах деятельности компании. В такой системе фактическая себестоимость сравнивается с тем ее значением, какое должно было быть при фактическом уровне деятельности.

Большинство систем расчета нормативной себестоимости включают пять основных этапов:

1. Определение нормативных затрат на единицу продукции для каждого центра ответственности.
2. Расчет нормативной себестоимости фактического объема продукции для каждого центра ответственности.
3. Документальная запись фактических затрат по каждому центру ответственности.
4. Сравнение нормативных и фактических затрат, анализ отклонений и внесение корректирующих изменений.
5. Контроль и корректировка нормативов для внесения изменений в конкретных расчетах и ценах.

По первичным данным, собранным по методу расчета себестоимости по нормативным затратам, можно анализировать и прогнозировать распределение затрат при помощи различных экономико-математических методов и моделей. Предполагается включение в систему следующих моделей:

- модель Брауна и Хольта (модель экспоненциального сглаживания).
- авторегрессионные модели прогнозирования.
- модель со скользящим средним.

Модель Брауна может отображать развитие не только в виде линейной тенденции, но также в виде изменяющейся параболической тенденции. Модель первого порядка имеет вид: $Y(t+k) = A_0 + A_1 * k$. Здесь коэффициент A_0 - значение, близкое к последнему уровню, и представляет как бы закономерную составляющую этого уровня. Коэффициент A_1 определяет прирост, сформировавшийся в основном к концу периода наблюдений.

Для авторегрессионных моделей вида $Y_t = \beta_0 + \beta_1 Y_{t-1} + \varepsilon_t$ прогнозы зависят от значений данных, наблюдаемых в предыдущие промежутки времени.

В этой модели прогноз находится как функция от предыдущих значений временного ряда.

Модель со скользящим средним порядка q задается следующим уравнением.

$$Y_t = \mu + \varepsilon_t - \omega_1 \varepsilon_{t-1} - \omega_2 \varepsilon_{t-2} - \dots - \omega_q \varepsilon_{t-q}$$

где Y_t - отклик (зависимая переменная) в момент времени t ;

μ - постоянное среднее процесса;

$\omega_1, \omega_2, \dots, \omega_q$ - оцениваемые коэффициенты;

ε_t ошибки в предыдущие периоды времени, которые в момент t включены в отклик Y_t .

Модели со скользящим средним (МА) дают прогноз значения функции Y_t , основываясь на линейной комбинации ограниченного числа прошлых ошибок Y_t , тогда как авторегрессионные модели (АР) дают прогноз, на основании линейной функции аппроксимации ограниченного числа прошлых значений.

Включение алгоритмов на основе перечисленных моделей в автоматизированную систему обработки экономической информации позволит прогнозировать основные категории затрат, существенно уменьшить затраты, связанные с обработкой данных и улучшить связь между разными подразделениями предприятия с использованием общей базы данных.

ЛИТЕРАТУРА

- [1] За ред. В.Д. Базидевича *Економічна теорія: Політекономія* // Підручник. – 5-те вид. – 2006. – 615 с.
- [2] В.В. Федосеев, А.Н. Гармаш, Д.М. Дайитбегов и др.; под ред. В.В. Федосеева. *Економіко-математическіе методи и прикладніе модели* // Учеб. пособие для вузов. – 2000. – 391 с..

Полиномиальная реализация частичных булевых функций

Клеймёнова Ю.Л.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (500и)

ВВЕДЕНИЕ

Данная статья посвящена проблеме полиномиальной реализации частичных булевых функций, которая является частным случаем задачи обучения по прецедентам [1], где в качестве множества объектов выбирается $X = B^n = \underbrace{B_2 \times B_2 \times \dots \times B_2}_n$, множество ответов $Y = B_2 = \{0, 1\}$, а множество A решающих правил является классом полиномов Жегалкина.

Данная задача является одной из основных задач кибернетики. К задачам обучения по прецедентам относятся такие задачи, как задача распознавания образов, задача классификации (задача медицинской диагностики, распознавание отдельных рукописных символов, задача классификации месторождений полезных ископаемых и т.д.), задача восстановления регрессии и задача прогнозирования (прогнозирование потребительского спроса, цен на рынке). На сегодняшний день значение такого рода задач очень высоко, поэтому растет потребность в приложениях, в рамках которых решаются подобные задачи.

В данной статье рассмотрим возможность использования алгоритмов, которые ранее применялись для реализации частичных булевых функций, описывающих некоторые AND/EXOR схемы минимальным полиномом, в задаче обучения по прецедентам.

ПОЛИНОМИАЛЬНАЯ РЕАЛИЗАЦИЯ ЧАСТИЧНЫХ БУЛЕВЫХ ФУНКЦИЙ.

Если значения булевой функции заданы лишь на некоторых наборах, то ее называют *частичной*.

Если функция частичная, то значения функции на неопределенных наборах (don't cares – безразличные наборы) в общем случае могут быть произвольными. Если число безразличных наборов k , то существует 2^k различных доопределений функции, и каждому из них соответствует свой полином (если этот полином является канонической формой, например как полином Жегалкина). В этом случае возникает задача оптимизации, то есть из всех возможных полиномов необходимо выбрать самый «наилучший».

Если рассматривать задачу обучения по прецедентам, то выбор «наилучшего» полинома будет определяться с помощью оценки качества, существует несколько основных типов таких оценок:

- *оценка эмпирического риска*, определяется с помощью функции потерь, которая характеризует величину отклонения полученного значения от правильного ответа, и весовой функции, которая определяет степень важности объекта x (признака) [3];
- *оценка VCD (Vapnik-Chervonenkis Dimension)*, также ее называют ёмкостью. Это статистическая оценка, основным ее недостатком считается то, что в процессе ее вычисления не учитываются свойства объектов обучающей выборки, что позволило бы существенно сузить пространство допустимых решений [2];
- существует несколько видов оценок сложности при решении задачи выбора модели методом *скользящего контроля (CV)* [1];
- понятие *колмогоровской сложности* является основным при решении задачи выбора модели по принципу минимума длины описания [5];

Для оценки полиномов Жегалкина будем использовать ёмкость класса решающих правил (VCD). Полученная в [1, С. 64-65] оценка делает актуальной задачу построения полиномов Жегалкина с наименьшим числом конъюнкций и литералов в них.

Данная задача также актуальна и для построения оптимальных AND/EXOR схем [4].

Для решения задачи полиномиальной реализации частичной булевой функции существует несколько алгоритмов, рассмотрим некоторые из них:

1. Построение полинома Жегалкина для частичных булевых функций методом прямого перебора.

Очевидно, что в общем случае при решении поставленной задачи данным способом имеется решение всегда. Данный метод был предложен в [7], и усовершенствован в [6], где минимизировали время получения очередного полинома.

Данный метод алгоритмичен и легко программируется, он у него есть существенный недостаток: его трудоемкость быстро растет с увеличением числа переменных в рассматриваемых частичных функциях. уже при $n = 6$ реализация алгоритма становится проблематичной, если предположить, что функция определена ровно на половине наборов, то $k = 32$, а вот количество полиномов $2^k = 4,3 \cdot 10^9$.

Именно из-за этого недостатка был разработан более эффективный метод, основанный на решении системы линейных логических уравнений.

2. Построение полинома Жегалкина для частичных булевых функций методом решения системы линейных логических уравнений и лестничным алгоритмом.

Метод поиска кратчайшего решения системы логических уравнений основан на игнорировании всех безразличных наборов k . Задача сводится к рассмотрению системы m линейных уравнений, соответствующих наборам, на которых функция определена, и решается методом поиска минимальных корней этой системы.

Основная идея данного метода заключается в следующем. Выделить q свободных переменных, которые могут принимать произвольные значения, число их различных наборов равно 2^q . Перебирая по очереди значения данных q свободных переменных (в порядке неубывания их весов) и оценивая длину s представляемых ими решений x , выбрать из этих решений кратчайшее.

Поставленную задачу также можно решить методом обхода дерева поиска, реализующим сокращенный перебор возможных решений при поиске оптимального из них, который называется *лестничным алгоритмом* минимизации полинома Жегалкина [4].

Данный алгоритм существенно сокращает объем производимого перебора и особенно эффективен для слабо определенных булевых функций, поскольку число конъюнкций в минимальном полиноме никогда не превышает число наборов, на которых определена функция.

При экспериментальных испытаниях алгоритма поиска кратчайшего решения системы логических уравнений (алгоритм А), учитывая то, что m —число уравнений, p —число неизвестных и r —ранг системы, также условие совместности и линейной независимости уравнений, которое накладывает ограничение на параметры $m = r \leq p$, было выявлено, что на скорость поиска кратчайшего решения влияют не параметры m , p и r , а их соотношения, особенно разность $q = p - r$, определяющая число решений системы.

В [4] доказано, что алгоритм А проигрывает по быстродействию лестничному алгоритму (алгоритм В). Но в случае коротких полиномов, даже при большом числе p , алгоритм А оптимальнее алгоритма В за счет своей простоты.

Анализ лестничного алгоритма показывает, что он существенно сокращает перебор всех полиномов, реализующих частичную булеву функцию, также он очень эффективен за счет высокого быстродействия, но с ростом n даже он становится практически нереализуемым.

Поэтому в некоторых случаях удобнее воспользоваться приближенным алгоритмом.

3. Приближенный алгоритм построения полинома Жегалкина для частичных булевых функций.

Данный алгоритм обеспечивает получение «наилучшего» решения за приемлемое время для частичных булевых функций многих переменных. Он работает с табличным представлением рассматриваемых функций, а сложность вычисления не превышает $O(m^2n)$.

Приближенный алгоритм является рекурсивным и заключается в том, чтобы последовательно получать конъюнкции на каждом шаге, модифицируя вектор значений нашей функции f , пока он не будет состоять только из нулей. Число таких конъюнкций не будет превышать m , это является верхней достижимой оценкой числа конъюнкций в полиномах, получаемых приближенным алгоритмом. Но нам более интересна нижняя оценка сложности получаемых полиномов. Эту оценку нельзя получить аналитическим путем, поэтому оценка качества полиномов, полученных приближенным алгоритмом, производится путем сравнения их с полиномами, полученными точным лестничным алгоритмом для тех же булевых функций.

В [4] рассматривается сравнение результатов, полученных приближенным и лестничным алгоритмами. То есть для каждого n —числа переменных функции получены верхние и нижние оценки сложности и затраченное время t . Исходя из этого сравнения можно сделать вывод, что приближенный алгоритм получает решение намного быстрее, чем лестничный, но при этом нижняя оценка меньше (тем самым "выгоднее") у второго алгоритма.

ЗАКЛЮЧЕНИЕ

В статье рассмотрены основные алгоритмы построения полиномов Жегалкина с наименьшим числом конъюнкций и литералов, что является актуальным при решении задачи обучения по прецедентам и построения AND/EXOR схем.

ЛИТЕРАТУРА

- [1] Анафиев А.С. Теория шаблонов в задачах обучения по прецедентам и выбора моделей. // Диссертация на соискание ученой степени кандидата физико-математических наук, 2007 г.
- [2] Анафиев А.С. Оценка VCD для k —параметрических шаблонов $t[+; \cdot]$. // Таврический вестник математики и информатики. – 2005. – №2. – С.102-108.
- [3] Воронцов К.В. Комбинаторные оценки качества обучения по прецедентам. // Доклады РАН. – 2004. – Т.394, №2 – С.175–178.

- [4] *Закревский А.Р., Торопов Н.Р.* Полиномиальная реализация частичных булевых функций и систем. // Едиториал УРСС, 2003 г., т. 62, № 2, 1948 г.
- [5] *Колмогоров А.Н.* Теория информации и теория алгоритмов. –М.: Наука, 1987. – С.304.
- [6] *Супрун В.П.* Минимизация частичных булевых функций в классе полиномов ридд-Мюллера //Автоматизация проектирования дискретных систем. - Минск: Ин-т техн. кибернетики АН Беларуси, 1995. - С. 34-47.
- [7] *Green D.H.* Reed-Muller expansions of incompletely specified functions //IEE Proc.E, Computers & Digital Techniques, 1987, vol. 134, pp. 228-236.

Современные методы упорядочения разреженных матриц для МКЭ

Косенюк Н. Г.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500И)

Одна из существенных проблем, проявляющихся при реализации расчётов методом конечных элементов (МКЭ), связана с решением систем линейных алгебраических уравнений большого порядка. Учитывая тенденцию неуклонного роста размерности задач (порядок систем МКЭ уравнений достигает 200 000-600 000 уравнений и более), а также тот факт, что при поиске приемлемого конструкторского решения приходится многократно вносить изменения в расчетную модель и, следовательно, каждый раз выполнять МКЭ-анализ заново, возникает потребность в разработке и внедрении в расчетные программные комплексы высокоэффективных методов решения систем МКЭ-уравнений. Особенности этих уравнений являются симметричность, положительная определённость и разреженность.

Рассмотрим многофронтальный метод факторизации [1, 2, 3] разреженных матриц и его способы повышения эффективности.

Ответственными моментами при создании эффективного прямого метода решения систем МКЭ-уравнений являются упорядочение, рациональная работа с кэш-памятью и шиной, а также организация работы с данными.

Упорядочение состоит в перенумерации уравнений и обычно преследует две цели:

1) уменьшить количество заполнений, возникающих при факторизации матрицы. [2].

2) разделить данные на примерно равнозначные с точки зрения вычислительных затрат порции с целью эффективного распараллеливания операций в параллельных вычислительных системах.

Решение задачи оптимизации количества заполнений в точной постановке требует не меньших вычислительных усилий, чем решение данной МКЭ-задачи со случайным образом расставленными уравнениями (отсутствие упорядочения). Поэтому на практике применяют эвристические алгоритмы упорядочения. Начиная с 70-х годов и до середины 90-х в МКЭ-программах как правило применялись профильные методы решения систем линейных алгебраических уравнений с разреженными матрицами [2, 3]. При этом задачей упорядочения являлась минимизация ширины профиля. Чаще всего для этой

цели применялся обратный алгоритм Катхилла-Макки (RCM [2, 3]), а позднее – более эффективный метод Слоана [5].

С середины 90-х годов в коммерческие МКЭ-программы интенсивно внедряются методы, тонко учитывающие разреженную структуру матрицы жесткости (sparse direct solvers). При этом чаще всего применяется упорядочение алгоритмом минимальной степени и методом вложенных сечений [2, 3, 6].

Алгоритм минимальной степени на каждом шаге факторизации в качестве ведущего выбирает то уравнение, которое содержит наименьшее количество ненулевых коэффициентов.

Метод вложенных сечений рассекает исходную конструкцию на две (или более) подконструкции, примерно равных по количеству узлов. Затем для каждой из подконструкций процедура вложенных бисекций повторяется снова и т.д. В результате получается последовательность рекурсивно вложенных подсистем.

Оказалось, что для вытянутых в одном направлении конструкций лучшие результаты обычно дает алгоритм минимальной степени, а для конструкций, подобных кубу, – метод вложенных сечений [6]. Применение смешанного подхода, в котором сначала несколько шагов выполняется методом вложенных сечений, а внутри каждой подконструкции используется упорядочение алгоритмом минимальной степени, в сочетании с работой на огрубленном графе смежности позволило получить устойчивый по эффективности для широкого класса конструкций многоуровневый метод упорядочения [7, 8].

В процессе факторизации одновременно образуются несколько фронтов, то метод называют многофронтальным. Такой подход оказывается эффективным как для параллельных вычислений, так и для обычных компьютеров.

ЛИТЕРАТУРА

- [1] Деммель Дж *Вычислительная линейная алгебра. Теория и приложения* – М.: Мир, 2001. – 429 с.
- [2] Джордж А., Лю Дж *Численное решение больших разреженных систем уравнений*. – М.: Мир, 1984. – 333 с.
- [3] Писсанецки С. *Технология разреженных матриц*. – М.: Мир, 1988. – 411 с.
- [4] Ashcraft C., Liu J. W.-H. *Robust Ordering of Sparse Matrices Using Multisection*. – Technical Report CS 96-01, Department of Computer Science, York University, Ontario, Canada, 1996.
- [5] Duff I.S., Reid J.K., Scott J.A. *The use of profile reduction algorithms with a frontal code*. – Int. J. Numer. Meth. Eng. –1989 – 28 – P. 2555-2568.
- [6] George, A., Liu, J. W.-H. *The Evolution of the Minimum Degree Ordering Algorithm* – SIAM Rev. 31, 1-19 (March 1989).

- [7] Karypis G., Kumar V. *A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs* – Technical Report TR 95-035, Department of Computer Science, University of Minnesota, Minneapolis, 1995.
- [8] Karypis G., Kumar V. *METIS: Unstructured Graph Partitioning and Sparse Matrix Ordering System* – Technical report, Department of Computer Science, University of Minnesota, Minneapolis, 1995.

Применение теории Спрага-Гранди для обобщенной игры Кейлс

Кудла А. А.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

Классическая игра «Кейлс», которая довольно распространена, хотя и малоизвестна, была разработана английским математиком Даденеом в начале XX века. Ее смысл довольно прост: выкладывается в ряд одиннадцать спичек, отдельно кладется еще одна. Каждый из двух игроков за ход может взять одну либо две любых спички. Выигрывает тот, кто забирает последнюю спичку.

Ключевое отличие этой игры от другой классической игры «Ним» в том, что количество объектов может увеличиваться. Проще говоря, игрок своим ходом может увеличить число кучек со спичками. Это не просто дает дополнительные возможности – это усложняет игру до предела, позволяя варьировать возможные варианты игр, стратегии в которых могут быть в корне отличными от стратегии в изначальной игре. Классическая «Кейлс» - только верхушка айсберга, скрывающая под собой огромное число игр и идей. Поэтому предлагается именовать «Кейлс» всякую игру, соответствующую следующему определению.

Определение 12. *Игрой кейлс называется любая игра для двух игроков, которые могут делать следующие ходы: в позиции, которая представляет собой вектор из натуральных чисел, из любого из них можно вычесть целое число из некоторого интервала. Кроме того, по желанию игрока координата вектора, с которой производится действие, может быть предварительно разбита на любую сумму натуральных чисел, которые станут полноценными координатами вектора позиции, при этом не нарушая двух правил: 1) координаты вектора не могут быть отрицательны; 2) ход обязательно должен изменить позицию.*

Будем обозначать определенную игру $K_{(a,b)}^r$, где a, b – границы интервала, в пределах которого разрешено брать делать ход, r – подмножество множества правил, который может быть представлен битовым вектором, каждая координата которого означает, применяется или нет данное правило в данной игре. Разнообразные вариации этих правил будут рассмотрены ниже. Поскольку особо будет рассматриваться случай, когда $a = 1$, для упрощения записи это число будет опускаться. Например, классический даденеевский кейлс запишется как K_2 .

Позицией в игре будет являться вектор $X = \{x_1, x_2, \dots, x_n\}$, к которому будет применяться игра. Будем считать, что координаты вектора упорядочены по убыванию, т.е. $i > j \rightarrow x_i \leq x_j$. Решить игру кейлс значит построить алгоритм, который для заданной игры кейлс вычисляет исход игры в любой позиции при наилучшей игре соперников. Следует отметить, что оптимальные алгоритмы могут различаться в зависимости от правил игры.

Первое существенное различие между различными играми - "вопрос о последней спичке", т.е. исход игры для игрока, забравшего последнюю спичку. В зависимости от того, выигрывает либо проигрывает этот игрок, проведем разделение на W -кейлс и L -кейлс, обозначать которые будем как WK и LK соответственно. Заметим, что для WK условие победы возможно переформулировать как "проигрывает тот, кто не может сделать ход", что потребуется нам для случая, когда $a > 1$. Для LK имеет смысл ввести ограничение $x_i \leq 1$, либо доопределить исход, в котором ни один игрок не может сделать ход. В рамках моей работы внимание будет уделено именно первому варианту.

Существенным отличием данной игры от классического кейлс является вариант задачи с отрицательным a . Сразу введем ограничение, согласно которому $|a| < |b|$ - иначе любой ход одного игрока, не производящий увеличение размерности вектора позиции, может быть "отменен" другим игроком путем добавления к той же координате такого же числа, какое вычел первый игрок. Данный кейлс можно назвать "шведским по аналогии с так называемыми "шведскими" шахматами. Три правила, которые серьезно отличают разные варианты подобного кейлс - правило "нулевого хода", правило соединения и правило увеличения.

1) Правило нулевого хода - разрешение игроку увеличить число координат вектора, не производя при этом вычитания (проще говоря, разбить одну линию спичек на две). Приведенным выше правилам оно не противоречит, т.к. позиция изменяется.

2) Правило соединения - прямая противоположность предыдущему. Это разрешение уменьшить число координат вектора, соединив их. Возможна вариация, когда для совершения этого действия необходимо добавить как минимум одну спичку.

3) Правило увеличения - разрешение игроку увеличить сумму всех координат вектора. Т.е. это возможность перейти в позицию, в которой спичек больше, чем в начальной. Сразу оговоримся, что в играх, которые мы будем рассматривать, это правило в список действующих включено не будет, т.к. оно существенно усложняет перебор позиций, поскольку дерево решений становится фактически бесконечным.

Все три перечисленных выше правила описываются вектором r (от английского Rules). Мы будем основываться на варианте $(0, 0, 0)$ - т.е. наиболее простом, чтобы далее перейти к более сложным. В любом случае, нам нужно будет добавить во множество исходов ничейный. Необходимость этого шага можно пояснить следующим примером. Допустим, что в некотором шведском кейлс из позиции $\{x_1, x_2, \dots, x_n\}$ все ходы ведут в выигрышные позиции, кроме одной, допустим, $\{x_1, x_2 + k, \dots, x_n\}$. В этой же позиции ситуация аналогична - все позиции, в которые можно походить из нее, выигрышны, кроме, конечно, $\{x_1, x_2, \dots, x_n\}$. Таким образом, игроки, чтобы не отдать победу противнику, будут просто чередовать эти две позиции. Поэтому целесообразным является введение правила "троекратного повторения ходов", аналогичное шахматному. В случае, если некоторая позиция повторится три раза при ходе одного и того же игрока, игра считается окончившейся вничью.

Таким образом, поставленная научная задача сформулирована так: необходимо построить эффективные алгоритмы, учитывающие особенности каждого возможного варианта игры. Кроме того, предполагается получить некоторые статистические результаты, например, "коэффициент неудачи процент среди позиций длины L (длиной позиции называется сумма всех элементов позиции) проигранных позиций и зависимость динамики изменения этого коэффициента от значений a и b , а также от правил.

Предполагается, что программная реализация будет изначально использовать простые, но надежные методы, а уже впоследствии, после получения результатов, на их основе будут построены более сложные и эффективные алгоритмы.

ЛИТЕРАТУРА

- [1] Kenyon, John Charles *Nim-like games and the Sprague-Grundy theory* //University of Calgary, 1967. – С. 73-76.
- [2] Elwyn R . Berlekamp, John Horton Conway, Richard K. Guy. *Winning Ways for Your Mathematical Plays.* –A K Peters, Ltd., 2001. – 276 с.

Оценивание дипломных проектов с использованием нечётной логики

Кузнецова О. В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. Вернадского
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА (500-И)

Каждый преподаватель сталкивался с проблемой оценивания дипломных проектов студентов, но четкой системы, которая позволила бы ему принять это непростое решение со всей объективностью, не существует до сих пор. Однако всякая задача требует своего решения.

Задача ставилась следующим образом: создать программу, которая на основе обучающей выборки, то есть принятых ранее решений по оцениванию, занесенных в базу знаний, позволяет оценить дипломную работу студента на основе различных критериев. Необходимо было создать набор правил, которые определяли бы оценку дипломного проекта.

Задача оценки качества дипломного проектирования принадлежит к широкому классу задач оценки качества интеллектуальной деятельности. Типичными представителями таких видов деятельности являются: подготовка статей, диссертаций, инновационных проектов и другой технической продукции. Использование нечеткой логики для решения такого рода задач позволяет максимально приблизить математическую модель оценки качества к логике рассуждений квалифицированных специалистов, принимающих оценочные решения.

Построение нечетких экспертных систем, которые основаны на предложенной методике, дает возможность не только оценить качество интеллектуальной деятельности, но и создает условия для ее проектирования, т.е. такого управления частными показателями, при котором интегральный показатель качества будет принимать желаемое значение.

1. ОЦЕНИВАНИЕ ДИПЛОМНЫХ ПРОЕКТОВ С ИСПОЛЬЗОВАНИЕМ НЕЧЁТКОЙ ЛОГИКИ

Усовершенствование системы оценки качества дипломного проектирования представляет интерес в двух аспектах: информационном и управленческом. Информационный аспект связан с возможностью создания автоматизированных банков данных, содержащих информацию о личных и профессиональных качествах студента-дипломника. Управленческий аспект состоит в

том, что во время дипломирования студент-дипломник должен иметь возможность оперативно прогнозировать свою будущую оценку и вносить коррективы для ее улучшения.

Известная методика оценки дипломных проектов и работ базируется на скаляризации векторного критерия качества при помощи весовых коэффициентов. Последние выбираются экспертно, отображая важность каждого частного показателя качества и его вклад в интегральный показатель, который оценивается по традиционной пяти бальной системе.

Не останавливаясь подробно на известных из теории многокритериальной оценки трудностях в подборе весовых коэффициентов, следует признать, что главным недостатком методики является то, что она весьма далека от реальной схемы рассуждений, которыми пользуются квалифицированные специалисты (члены комиссии) при оценке работы дипломника. Эти рассуждения выглядят примерно так:

ЕСЛИ работа актуальна, выполнена самостоятельно, решения глубоко обоснованы, оформление отвечает стандартам, результаты пригодны к внедрению, во время защиты даются четкие ответы на вопросы,

ТО дипломник заслуживает отличной оценки.

Рассуждения такого рода, зафиксированные в виде словесных инструкций (нечетких алгоритмов) есть, как правило, на всех выпускающих кафедрах. Поэтому целесообразно их непосредственное использование для оценки качества дипломного проектирования.

2. АЛГОРИТМ РЕШЕНИЯ

Для решения исходной задачи было решено применить классический алгоритм обучения персептрона. Получаем некоторое множество прецедентов, которые принадлежат одному из трех классов, соответствующих оценкам "3", "4" и "5". На основе этой обучающей выборки мы находим коэффициенты гиперплоскостей, отделяющих классы друг от друга, с помощью правила "подкрепления наказания". Теперь с помощью полученных функций мы можем классифицировать дипломный проект, не входящий в базу знаний.

Следует заметить, что при построении нашего алгоритма мы исходим из монотонности функции оценивания, иначе говоря, если один набор критериев по правилу сравнения векторов больше другого, то выставленная за него оценка не ниже второй. Это объясняется самой простой логикой и здравым смыслом, и в то же время является обоснованием выбора данного алгоритма, поскольку в таком случае множества классов обучающей выборки разделимы.

3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

Данный алгоритм реализован программно в среде Delphi.

ЛИТЕРАТУРА

- [1] *Леоненков А.* Нечеткое моделирование в среде MatLab и fuzzyTECH. - СПб.: БХВ-Петербург, 2003. - 736 с.
- [2] *Ротштейн А.П.* Интеллектуальные технологии идентификации. - Электронный ресурс: <http://matlab.exponenta.ru/>

Малые колебания системы идеальных капиллярных жидкостей

Лившиц А.И.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА МАТЕМАТИЧЕСКОГО АНАЛИЗА (ГРУППА 500)

В данной работе рассматривается задача о малых колебаниях системы идеальных капиллярных жидкостей. Методом разделения переменных решаются задачи для цилиндрического сосуда, полностью заполненного двумя и тремя жидкостями. Находятся уравнения для нахождения собственных значений λ . Показано, что решения спектральной задачи для трех жидкостей асимптотически сходятся к решениям задач для двух жидкостей.

Пусть замкнутый неподвижный сосуд полностью заполнен несколькими идеальными несжимаемыми жидкостями, занимающими в состоянии равновесия области $\Omega_1, \Omega_2, \dots, \Omega_{m+1}$. Предположим, что эти области расположены подобно слоеному пирогу, так что Ω_1 контактирует (кроме стенки сосуда) только с Ω_2 по некоторой поверхности Γ_1 ; Ω_2 – с Ω_1 и с Ω_3 (по Γ_1 и Γ_2) и т.д., каждая Γ_k контактирует с поверхностью сосуда по некоторой линии γ_k , причем эти линии не имеют попарно общих точек. Пусть ρ_k – плотность жидкости в Ω_k , σ_k – коэффициент поверхностного натяжения на Γ_k , α_k – угол смачивания k -й жидкости вдоль γ_k . Допустим, что на систему действует однородное гравитационное поле с массовой плотностью потенциала gz . Вектор нормали \vec{n} на Γ_k направим из Ω_k в Ω_{k+1} , N_k – отклонения от Γ_k по нормали \vec{n} .

Рассмотрим задачу о малых свободных колебаниях такой системы. Движение каждой жидкости считаем потенциальным с потенциалом скорости $\Phi_k(x, t)$ ($u_k = -\nabla\Phi_k$); имеем

$$\Delta\Phi_k = 0 \text{ (в } \Omega_k), \quad \frac{\partial\Phi_k}{\partial n} = 0 \text{ (на } \Sigma_k) \quad (k = 1, 2, \dots, m+1), \quad (1)$$

где Σ_k – часть Σ , контактирующая с Ω_k . Кинематические условия приобретают вид

$$\frac{\partial N_k}{\partial t} = -\frac{\partial\Phi_k}{\partial n} \Big|_{\Gamma_k} = -\frac{\partial\Phi_{k+1}}{\partial n} \Big|_{\Gamma_k} \quad (k = 1, 2, \dots, m), \quad (2)$$

а динамические

$$\rho_k \frac{\partial\Phi_k}{\partial t} - \rho_{k+1} \frac{\partial\Phi_{k+1}}{\partial t} - \sigma_k (a_k N_k - \Delta_{\Gamma_k} N_k) = \psi_k(t) \text{ (на } \Gamma_k),$$

где $\psi_k(t)$ – произвольные функции времени,

$$a_k = \frac{(\rho_k - \rho_{k+1})g}{\sigma_k} \cos(\widehat{\vec{n}, \vec{z}}) - k_1^2 - k_2^2 \text{ (на } \Gamma_k).$$

Из (1) и (2) следует

$$\int_{\Gamma_k} N_k d\Gamma_k = 0, \quad \int_{\Gamma_k} \frac{\partial \Phi_k}{\partial n} d\Gamma_k = \int_{\Gamma_k} \frac{\partial \Phi_{k+1}}{\partial n} d\Gamma_k = 0 \quad (k = 1, 2, \dots, m).$$

На линиях γ_k должны выполняться условия

$$\frac{\partial N_k}{\partial e_k} + \chi_k N_k = 0 \quad (\text{на } \gamma_k; \quad k = 1, 2, \dots, m),$$

где $\partial/\partial e_k$ – производная по внешней нормали \vec{e}_k к контуру γ_k в плоскости, касательной к Σ_k . Здесь

$$\chi_k = (k_\Gamma \cdot \cos \delta - k_s) / \sin \delta$$

– функция, определенная на γ_k , k_Γ и k_s – кривизны линий, которые получаются в результате сечения поверхностей Γ_k и Σ_k плоскостью, перпендикулярной к γ_k , причем вычисленные на γ_k .

Далее рассмотрим сосуд, ограниченный цилиндрической поверхностью с вертикальными образующими, а снизу – плоским горизонтальным дном. Углы смачивания $\alpha_k = \frac{\pi}{2}$. Считаем, что решения начально-краевой задачи осциллируют во времени – зависят от времени по гармоническому закону $\exp(i\omega t)$, где ω – неизвестная частота колебаний. Полагая

$$N_k = N_k(x)e^{i\omega t}, \quad \Phi_k = \Phi_k(x)e^{i\omega t},$$

и исключая функции $N_k(x)$, получаем следующую задачу на собственные значения (так как $k_1 = k_2 = 0$, то $a_k = \frac{(\rho_k - \rho_{k+1})g}{\sigma_k}$, $\chi_k \equiv 0$):

$$\begin{cases} \Delta \Phi_k = 0 \quad (\text{в } \Omega_k), \quad \frac{\partial \Phi_k}{\partial n} = 0 \quad (\text{на } \Sigma_k) \quad (k = 1, 2, \dots, m+1), \\ \lambda(\rho_k \Phi_k - \rho_{k+1} \Phi_{k+1}) = -\sigma_k \Delta_{\Gamma_k} \frac{\partial \Phi_k}{\partial n} + (\rho_k - \rho_{k+1})g \frac{\partial \Phi_k}{\partial n} \quad (\text{на } \Gamma_k), \\ \frac{\partial \Phi_k}{\partial n} = \frac{\partial \Phi_{k+1}}{\partial n} \quad (\text{на } \Gamma_k), \quad \int_{\Gamma_k} \frac{\partial \Phi_k}{\partial n} d\Gamma_k = 0, \\ \frac{\partial}{\partial e_k} \left(\frac{\partial \Phi_k}{\partial n} \right) = 0 \quad (\text{на } \gamma_k) \quad (k = 1, 2, \dots, m). \end{cases} \quad (3)$$

Здесь $\lambda = \omega^2$ – собственное значение.

Для нахождения решения задачи (3) проведем разделение переменных, а именно:

$$\Phi_k(x, y, z) = X(x, y) \cdot Z_k(z).$$

Возникают следующие две задачи:

$$\begin{cases} \Delta_2 X - cX = 0 \quad (\text{на } \Gamma), \\ \frac{\partial X}{\partial e} = 0 \quad (\text{на } \partial\Gamma), \quad \int_{\Gamma} X d\Gamma = 0; \end{cases} \quad (4)$$

$$\begin{cases} Z_k'' + cZ_k = 0, \\ \frac{\partial Z_k}{\partial z} = \frac{\partial Z_{k+1}}{\partial z} \quad (\text{на } \Gamma_k), \\ \lambda(\rho_k \Phi_k - \rho_{k+1} \Phi_{k+1}) = -\sigma_k \Delta_{\Gamma_k} \frac{\partial \Phi_k}{\partial z} + (\rho_k - \rho_{k+1})g \frac{\partial \Phi_k}{\partial z} \quad (\text{на } \Gamma_k). \end{cases} \quad (5)$$

Далее показываем, что постоянная в (4) и (5) может быть лишь отрицательным числом, т.е. $c = -\mu_k^2$. В силу этого задачи (4) и (5) принимают следующий вид:

$$\begin{cases} \Delta_2 X + \mu_k^2 X = 0 \quad (\text{на } \Gamma), \\ \frac{\partial X}{\partial e} = 0 \quad (\text{на } \partial\Gamma), \quad \int_{\Gamma} X d\Gamma = 0; \end{cases} \quad (6)$$

$$\begin{cases} Z_k'' - \mu_k^2 Z_k = 0, \\ \frac{\partial Z_k}{\partial z} = \frac{\partial Z_{k+1}}{\partial z} \quad (\text{на } \Gamma_k), \\ \lambda(\rho_k \Phi_k - \rho_{k+1} \Phi_{k+1}) = -\sigma_k \Delta_{\Gamma_k} \frac{\partial \Phi_k}{\partial n} + (\rho_k - \rho_{k+1})g \frac{\partial \Phi_k}{\partial n} \quad (\text{на } \Gamma_k). \end{cases} \quad (7)$$

Известно, что задача (6) имеет дискретный спектр $\{\mu_k\}_{k=1}^{\infty} : 0 < \mu_1 \leq \mu_2 \leq \dots \leq \mu_k \leq \dots$, $\mu_k \rightarrow \infty$ и собственные функции $X_k(x, y)$, $k = 1, 2, \dots, \infty$.

Решим задачу (7) для цилиндрического сосуда, полностью заполненного двумя идеальными жидкостями. Они занимают области Ω_1 и Ω_2 , где $\Omega_1 = \Gamma \times (-h_1, 0)$, $\Omega_2 = \Gamma \times (0, h_2)$, Γ – поверхность раздела этих жидкостей.

В этом случае задача (7), учитывая дискретный спектр $\{\mu_k\}$, $k = 1, 2, \dots, \infty$, принимает вид:

$$\begin{cases} Z_k'' - \mu_k^2 Z_k = 0, \\ Z_1'(-h_1) = 0, \quad Z_2'(h_2) = 0, \\ Z_1'(0) = Z_2'(0), \\ \lambda(\rho_1 \Phi_1 - \rho_2 \Phi_2) = Z_1'(0)\beta_k, \end{cases} \quad (8)$$

где $\beta_k := \sigma_1 \mu_k^2 + (\rho_1 - \rho_2)g$.

Общее решение задачи (8) ищем в виде:

$$Z_k(z) = A_{1k} \cdot e^{\mu_k z} + A_{2k} \cdot e^{-\mu_k z}.$$

Из второй строчки системы находим:

$$Z_1(z) = C_{1k} \operatorname{ch}[\mu_k(z + h_1)] \quad (k = 1, 2, \dots, \infty),$$

$$Z_2(z) = C_{2k} \operatorname{ch}[\mu_k(z - h_2)] \quad (k = 1, 2, \dots, \infty).$$

Подставляем найденные решения в третью и четвертую строчки системы (8). Получаем однородную систему относительно двух уравнений с двумя неизвестными C_{1k} и C_{2k} , решая которую находим собственное значение:

$$\lambda_k^{(1)} = \frac{\mu_k \beta_k}{\rho_1 \operatorname{cth}(\mu_k h_1) + \rho_2 \operatorname{cth}(\mu_k h_2)}.$$

Пусть теперь две жидкости занимают области Ω_2 и Ω_3 , где $\Omega_2 = \Gamma \times (0, h_2)$, $\Omega_3 = \Gamma \times (h_2, h_3)$. В этом случае задача (7), учитывая дискретный спектр $\{\mu_k\}$, $k = 1, 2, \dots, \infty$, принимает вид:

$$\begin{cases} Z_k'' - \mu_k^2 Z_k = 0, \\ Z_2'(0) = 0, \quad Z_3'(h_3) = 0, \\ Z_2'(h_2) = Z_3'(h_2), \\ \lambda (\rho_2 \Phi_2 - \rho_3 \Phi_3) = Z_2'(h_2) \widetilde{\beta}_k, \end{cases} \quad (9)$$

где $\widetilde{\beta}_k := \sigma_2 \mu_k^2 + (\rho_2 - \rho_3)g$.

Решая задачу (9) аналогично задаче (8), находим такое собственное значение:

$$\lambda_k^{(2)} = \frac{\mu_k \widetilde{\beta}_k}{\rho_2 \operatorname{cth}(\mu_k h_2) + \rho_3 \operatorname{cth}[\mu_k (h_3 - h_2)]}.$$

Рассмотрим теперь задачу на собственные значения для цилиндрического сосуда, полностью заполненного тремя идеальными несжимаемыми жидкостями. Они занимают области Ω_1 , Ω_2 , Ω_3 , где $\Omega_1 = \Gamma \times (-h_1, 0)$, $\Omega_2 = \Gamma \times (0, h_2)$, $\Omega_3 = \Gamma \times (h_2, h_3)$.

Для этого случая после разделения переменных задача запишется в виде:

$$\begin{cases} Z_k'' - \mu_k^2 Z_k = 0, \\ Z_1'(-h_1) = 0, \quad Z_3'(h_3) = 0, \\ Z_1'(0) = Z_2'(0), \quad Z_2'(h_2) = Z_3'(h_2), \\ Z_1'(0) \beta_k = \lambda (\rho_1 z_1(0) - \rho_2 z_2(0)), \\ Z_2'(h_2) \widetilde{\beta}_k = \lambda (\rho_2 z_2(h_2) - \rho_3 z_3(h_2)), \end{cases} \quad (10)$$

где $\beta_k := \sigma_1 \mu_k^2 + (\rho_1 - \rho_2)g$, $\widetilde{\beta}_k := \sigma_2 \mu_k^2 + (\rho_2 - \rho_3)g$.

Решая задачу аналогично задачам (8) и (9), получаем характеристическое уравнение:

$$\begin{aligned} & \lambda^2 \left[\rho_1 \rho_3 \operatorname{cth}(\mu_k h_1) \operatorname{cth}[\mu_k (h_3 - h_2)] + \rho_2^2 + \rho_1 \rho_2 \operatorname{cth}(\mu_k h_1) \operatorname{cth}(\mu_k h_2) + \right. \\ & \left. + \rho_2 \rho_3 \operatorname{cth}(\mu_k h_2) \operatorname{cth}[\mu_k (h_3 - h_2)] \right] - \lambda \left[\rho_3 \mu_k \beta_k \operatorname{cth}[\mu_k (h_3 - h_2)] + \right. \end{aligned}$$

$$+ \rho_1 \mu_k \beta_l \operatorname{cth}(\mu_k h_1) + \mu_k \beta_k \rho_2 \operatorname{cth}(\mu_k h_2) + \rho_2 \mu_k \beta_l \operatorname{cth}(\mu_k h_2) \Big] + \mu_k^2 \beta_k \beta_l = 0. \quad (11)$$

Используя теорему Виета, составляем характеристическое уравнение, корнями которого являются собственные значения $\lambda_k^{(1)}$ и $\lambda_k^{(2)}$ задач (8) и (9) соответственно. Оказывается, что уравнение (11) отличается от составленного уравнения на бесконечно малую, стоящую при λ^2 , т.е. на величину

$$\rho_2^2 \left[\operatorname{cth}^2(\mu_k h_2) - 1 \right] \rightarrow 0 \quad (k \rightarrow \infty).$$

Отсюда делаем вывод, что при $k \rightarrow \infty$ корни уравнения (11) асимптотически сходятся к корням составленного уравнения и, следовательно,

$$\lambda_{1_k} \rightarrow \lambda_k^{(1)} \quad (k \rightarrow \infty),$$

$$\lambda_{2_k} \rightarrow \lambda_k^{(2)} \quad (k \rightarrow \infty),$$

где λ_{1_k} и λ_{2_k} – корни характеристического уравнения (11).

ЛИТЕРАТУРА

- [1] Мышкис А. Д., Бабский В. Г., Жуков М. Ю., Копачевский Н. Д. и др. *Методы решения задач гидромеханики для условий невесомости*. – Киев: Наук. думка, 1992. – 592 с.

Теорема Макки-Аренса для шкал локально выпуклых пространств

Менумерова С.Р.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА АЛГЕБРЫ И ФУНКЦИОНАЛЬНОГО АНАЛИЗА (ГРУППА 500)

Аннотация. Известный критерий Макки-Аренса согласованности топологии с двойственностью пространств обобщен на индуктивные шкалы пространств.

Классическая проблема Макки состоит в описании всех локально выпуклых топологий в пространстве E , согласованных с двойственностью $\langle E, F \rangle$, т.е. таких топологий $t(E)$, для которых справедлив векторный изоморфизм

$$(E, t(E))^* \cong F.$$

Решение проблемы Макки дает известная теорема Макки-Аренса [1], которую можно сформулировать в виде оценки снизу и сверху топологии $t(E)$.

Теорема 8. (Макки-Аренса). Пусть двойственность $\langle E, F \rangle$ отделима по F . Топология $t(E)$ в E согласована с двойственностью $\langle E, F \rangle$ в том и только в том случае, когда

$$\sigma(E, F) \preceq t(E) \preceq \tau(E, F), \quad (1)$$

где $\sigma(E, F)$ — слабая топология, а $\tau(E, F)$ — топология Макки в E относительно данной двойственности.

Напомним, что сходимость в слабой топологии $\sigma(E, F)$ есть сходимость относительно всех конечных наборов функционалов из F , а сходимость в топологии Макки $\tau(E, F)$ есть сходимость относительно всех слабо компактных абсолютно выпуклых подмножеств функционалов из F .

Поставим вопрос об аналогах проблемы Макки и теоремы Макки-Аренса для индуктивных шкал локально выпуклых пространств (ЛВП).

Введем вначале необходимые понятия.

Определение 13. Будем говорить, что задана индуктивная шкала (спектр) векторных пространств (ВП) $\{E_s\}_{s \in S}$, если S индуктивно упорядочено, и каждой паре индексов $s_1 \preceq s_2$ из S отвечает линейный оператор (вложение) $e_{s_2 s_1} : E_{s_1} \hookrightarrow E_{s_2}$ с условиями согласованности:

$$(s_1 \preceq s_2 \preceq s_3) \Rightarrow (e_{s_3 s_1} = e_{s_3 s_2} \circ e_{s_2 s_1}); \quad e_{ss} = I_{E_s}.$$

Обозначим $\vec{E} = (\{E_s\}_{s \in S}, \{e_{s_2 s_1}\}_{s_1 \preceq s_2})$, или коротко $\vec{E} = \{E_s\}_{s \in S}$.

Если пространства E_s — ЛВП, а вложения $e_{s_2 s_1}$ — непрерывные, то шкалу \vec{E} назовем индуктивной шкалой ЛВП.

Определение 14. Индуктивно упорядоченное множество S называется (локально) полной решеткой, если любое ограниченное сверху (снизу) подмножество S имеет супремум (инфимум) в S . Если $\vec{E} = \{E_s\}_{s \in S}$ — индуктивная шкала ВП и S — полная решетка, то назовем \vec{E} полной индуктивной шкалой.

Определение 15. Пусть $\vec{E} = \{E_s\}_{s \in S}$ и $\vec{F} = \{F^t\}_{t \in T}$ — полные индуктивные шкалы ВП. Будем говорить, что задана двойственность $\langle \vec{E}, \vec{F} \rangle$, если $\forall s \in S \exists t \in T$, для которых задана двойственность $\langle E_s, F^t \rangle$ с условиями согласованности

$$(s_1 \preceq s_2, t_1 \preceq t_2) \implies (\langle E_{s_1}, F^{t_1} \rangle = \langle e_{s_2 s_1}(E_{s_1}) \subset E_{s_2}, f^{t_2 t_1}(F^{t_1}) \subset F^{t_2} \rangle).$$

Двойственность $\langle \vec{E}, \vec{F} \rangle$ замкнута, если $\forall t \in T \exists s \in S$: определена двойственность $\langle E_s, F^t \rangle$. Двойственность $\langle \vec{E}, \vec{F} \rangle$ отделима по \vec{E} , если $\forall s \in S \forall 0 \neq x \in E_s \exists t \in T : \langle x, F^t \rangle \neq \{0\}$, и отделима по \vec{F} , если $\forall t \in T \forall 0 \neq y \in F^t \exists s \in S : \langle E_s, y \rangle \neq \{0\}$

Определение 16. Пусть $\vec{E} = \{E_s\}_{s \in S}$ — полная индуктивная шкала ВП, T — полная решетка. Назовем проективным оснащением шкалы \vec{E} систему локально выпуклых топологий

$$r(\vec{E}) = \{r^t(E_s)\}_{s \in S, t \in T},$$

удовлетворяющих условиям:

- 1) при любом фиксированном $s \in S$ система $\{r^t(E_s)\}_{t \in T}$ — проективная система топологий в E_s , т.е. $(t_1 \preceq t_2) \implies (E_s, r^{t_2}(E_s) \hookrightarrow (E_s, r^{t_1}(E_s))$;
- 2) при любом фиксированном $t \in T$ система $\{(E_s, r^t(E_s))\}_{s \in S}$ — индуктивная шкала ЛВП, т.е. $(s_1 \preceq s_2) \implies (E_{s_1}, r^t(E_{s_1}) \hookrightarrow (E_{s_2}, r^t(E_{s_2}))$.

Топологии проективных пределов $r(E_s) = \varprojlim_{t \in T} r^t(E_s)$ назовем топологиями,

порожденными оснащением $r(\vec{E})$. Будем писать $\vec{E}_r = (\vec{E}, r(\vec{E}))$.

Будем писать также $r_1(\vec{E}) \preceq r_2(\vec{E})$, если $r_1^t(E_s) \preceq r_2^t(E_s)$ ($\forall s \in S, t \in T$).

Приведем важные примеры оснащений.

Пример 1. Нормальное оснащение [2].

Пусть $\vec{E} = \{E_n\}_{n=1}^\infty$ — счетная индуктивная шкала пространств Фреше с определяющими системами полуном $\{\|\cdot\|_n^m\}_{m=1}^\infty$, отвечающих топологиям $\nu^m(E_n)$. Стандартное условие возрастания полуном $\|\cdot\|_n^m \leq \|\cdot\|_n^{m+1}$ ($\forall n \in \mathbb{N}$) обеспечивает выполнение условия 1) в определении 16. Потребуем также, чтобы $\|x\|_{n+1}^m \leq \|x\|_n^m$ при $x \in E_n$ ($\forall m \in \mathbb{N}$); тогда выполнено и условие 2).

Топологии, порожденные нормальным оснащением $\nu(\vec{E}) = \{\nu^m(E_n)\}_{m,n=1}^\infty$, совпадают с исходными топологиями пространств E_n .

Пример 2. *Оснащения, порожденные двойственностью.*

Пусть $\langle \vec{E}, \vec{F} \rangle$ — двойственность шкал $\vec{E} = \{E_s\}_{s \in S}$ и $\vec{F} = \{F^t\}_{t \in T}$. Тогда оснащения $\sigma(\vec{E}, \vec{F}) = \{\sigma(E_s, F^t)\}$ и $\tau(\vec{E}, \vec{F}) = \{\tau(E_s, F^t)\}$ удовлетворяют условиям определения 16. Назовем их, соответственно, *слабым оснащением* и *оснащением Макки* шкалы \vec{E} . Топологии, порожденные этими оснащениями, совпадают, соответственно, со слабыми топологиями $\sigma(E_s, F)$ и топологиями Макки $\tau(E_s, F)$ в E_s ($s \in S$), где $F = \varinjlim_{t \in T} F^t$.

Проективное оснащение позволяет ввести понятие проективного индекса функционала на шкале.

Определение 17. Пусть \vec{E}_r — оснащенная полная индуктивная шкала ЛВП, $\Lambda = \{\Lambda_s \in (E_s, r(E_s))^*\}$ — функционал (согласованное семейство функционалов) на \vec{E} с условиями согласованности $(s_1 \preceq s_2) \Rightarrow (\Lambda_{s_1} = \Lambda_{s_2} \circ e_{s_2 s_1})$. Будем писать $\Lambda \in E_r^*$.

Положим $\forall s \in S$:

$$\rho_\Lambda(s) = \inf\{t \in T \mid \Lambda_s \in (E_s, r^t(E_s))^*\}.$$

Функцию $\rho_\Lambda : S \rightarrow T$ назовем проективным индексом Λ относительно оснащения $r(\vec{E})$. Для каждого проективного индекса ρ положим

$$E_r^{*\rho} = \{\Lambda \in E_r^* \mid \rho_\Lambda \preceq \rho\}.$$

Тогда пространства $E_r^{*\rho}$ образуют полную индуктивную шкалу $(\vec{E}, r(\vec{E}))^* = \vec{E}_r^* = \{E_r^{*\rho}\}_\rho$, которую мы назовем проективным разложением пространства E_r^* .

Определение 18. Пусть $\langle \vec{E}, \vec{F} \rangle$ — двойственность полных индуктивных шкал \vec{E} и \vec{F} , $r(\vec{E})$ — проективное оснащение \vec{E} . Будем говорить, что оснащение $r(\vec{E})$ согласовано с двойственностью $\langle \vec{E}, \vec{F} \rangle$, если имеет место изоморфизм индуктивных шкал ВП:

$$\overrightarrow{(\vec{E}, r(\vec{E}))^*} \cong \vec{F}.$$

Сформулируем теперь обобщенную проблему Макки: для заданной двойственности $\langle \vec{E}, \vec{F} \rangle$ индуктивных шкал ВП описать все проективные оснащения $r(\vec{E})$ шкалы \vec{E} , согласованные с данной двойственностью.

Ниже приводится решение в случае нормального оснащения.

Теорема 9. (Обобщенная теорема Макки-Аренса). Пусть \vec{E} и \vec{F} — полные индуктивные шкалы ВП с инъективными вложениями (в индуктивные пределы), $\langle \vec{E}, \vec{F} \rangle$ — замкнутая двойственность, отделимая по \vec{F} , $\nu(\vec{E})$ — нормальное оснащение \vec{E} . Тогда оснащение $\nu(\vec{E})$ согласовано с двойственностью $\langle \vec{E}, \vec{F} \rangle$ в том и только в том случае, если выполнено обобщенное неравенство Макки-Аренса

$$\sigma(\vec{E}, \vec{F}) \preceq \nu(\vec{E}) \preceq \tau(\vec{E}, \vec{F}).$$

Заметим в заключение, что аналогичный результат можно получить и для двойственности индуктивной и проективной шкал пространств.

Выражаю признательность научному руководителю И.В.Орлову за постановку задачи и внимание к работе.

ЛИТЕРАТУРА

- [1] Шефер Х. *Топологические векторные пространства* // М.: "Мир", 1971. — 359 с.
- [2] Orlov I. V. *Normal functional indices and normal duality* // *Methods of Functional Analysis and Topology*. — 2002. — Vol.8. — no 3. — P. 61-71.

Разработка Web-приложений тестирования знаний для дистанционного обучения

Мосенцова Л.В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ(ГРУППА 500-П)

Одним из важных направлений развития образования является разработка эффективных методик контроля и тестирования знаний обучаемого. Особую актуальность эти методы приобретают в системе дистанционного обучения. Целью работы была разработка системы тестирования знаний [1],[2], простой в использовании как для тестируемого, так и для преподавателя (наполняющего систему тестами). Программа тестирования является динамическим Web-приложением, написанным на языке Java [3],[9],[8]. Для приложения была выбрана клиент - серверная модель [4]. Эта модель позволяет наиболее просто собирать результаты прохождения тестов и организовывать их централизованную обработку. В качестве Web - сервера используется Apache Tomcat [7], а базы данных - PostgreSQL . При разработке приложения использовались следующие Java- технологии: Spring [5], Hibernate [6], Velocity.

Система обладает следующими возможностями и свойствами:

- Позволяет добавлять/ удалять пользователей.
- В системе два вида пользователей: администраторы и рядовые пользователи. По умолчанию все новые пользователи рядовые, только администратор имеет право изменить их статус.
- В системе администратору разрешается добавлять/ удалять тесты.
- Рядовой пользователь может пройти тесты, просмотреть статистику по завершённому тесту, свой и чужой профиль.
- Каждый тест пользователь может пройти только один раз.
- В системе срабатывает защита, если пользователь пытается перейти на следующий вопрос, не выбрав вариант ответа.
- Система позволяет отслеживать всех пользователей сайта в нужный момент времени.
- Ведется протокол регистрации пользователей.
- Система разрешает включать в тест вопросы с возможностью выбора нескольких правильных вариантов.
- Система производит анализ результатов тестирования в соответствии с классической теорией тестов.
- Время тестов ограничено.

Все данные (о пользователях, о тестах и др.) хранятся в базе данных . Чтобы связать Java – объекты и данные в базе используется Hibernate - инструмент объектно-реляционного отображения данных для Java окружения, позволяющий не только отображать объекты Java классов в таблицы базы данных, но и поддерживает механизм запроса и поиска данных[11]. В приложении применяется модуль среды Spring Framework Spring Model-View-Controller (MVC)[10] . Присутствующая в нем логика шаблона MVC позволяет отделить данные от представления за счет введения промежуточного слоя преобразования называемого контроллером. Для представления данных служит Velocity-своеобразный движок для обработки шаблонов. Предназначается для отделения Java-кода от кода веб-страниц, генерирования веб-страниц, SQL-запросов, XML-документов. Приложение используется для проверки знаний по предмету "Базы данных".

ЛИТЕРАТУРА

- [1] Максимов А.В. Материалы рассылки “Тестирование. История, теория и практика.”(архивwww.subscribe.ru)
- [2] Frank B. Baker-The Basics Of Item Response Theory. ERIC Clearing-house. ISBN 1-886047-03-0
- [3] Bruce Eckel.Thinking in Java.Second edition. Revision 11
- [4] Budi Kurniawan.Java for the Web with Servlets, JSP, and EJB: A Developer’s Guide to J2EE Solutions.New Riders Publishing.April 12. 2002.0-7357-1195-X
- [5] Craig Walls,Ryan Breidenbach.Spring in Action.Second Edition.ISBN 1-933988-13-4
- [6] Christian Bauer.Gavin King. Hibernate in Action.ISBN 1932394-15-X
- [7] James Goodwill.Apache Jakarta-Tomcat.Apress © 2002 (237 pages).ISBN: 1893115364.
- [8] Greg M. Travis.JDK 1.4 Tutorial.ISBN 1-930110-66-9
- [9] Jason Hunter.Java Servlet Programming. 1-56592-391-X
- [10] Thomas Risberg.Spring MVC step-by-step
- [11] John O’Donahue. Java Database Programming Bible.John Wiley and Sons © 2002 (702 pages).ISBN:0764549243

Задача визуализации корневых деревьев

Москалев В.Ю.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

Решение задач визуализации информации является важным аспектом процесса разработки программного обеспечения. Задача визуализации графов и в частности деревьев на сегодняшний день – одна из сложнейших задач визуализации. Отсутствие единого алгоритма решения задачи визуализации корневых деревьев связано с большим разнообразием постановок этой задачи, начиная с выбора архитипа, наиболее изученными являются классический (Рис. 1) и блочный (Рис. 2) архитипы [1], и заканчивая такими требованиями к визуализации как централизация предков относительно потомков, конгруэнтное рисование для изоморфных поддеревьев и др. [2].

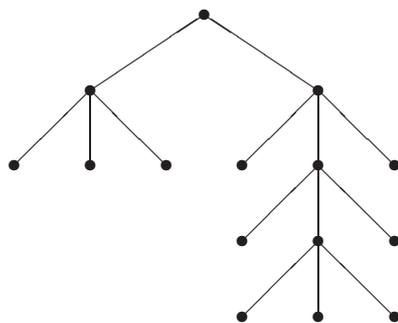


Рис. 1.

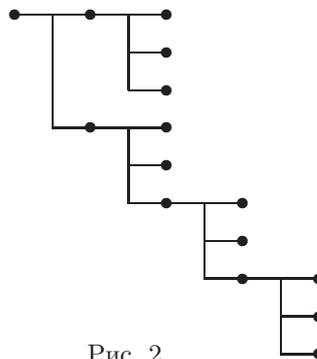


Рис. 2.

В настоящей работе для радиального архитипа визуализации корневых деревьев предлагается спецификация с более равномерным распределением вершин.

Опишем общую схему радиального типа визуализации. За основу построения корневого дерева берутся концентрические окружности по количеству слоев дерева. Корень дерева помещается в центр кругов. Следующим шагом внутренний круг делится на равные секторы, количество которых совпадает со степенью корневой вершины. Вершины первого слоя дерева располагаются равномерно на самой внутренней окружности. Вершины второго слоя располагаются равномерно на дугах, соответствующих предкам и т.д. Для графа G изоморфного графам на Рис. 1, 2 построим визуализацию по радиальному архитипу (Рис. 3).

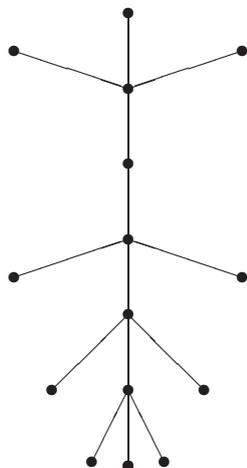


Рис. 3.

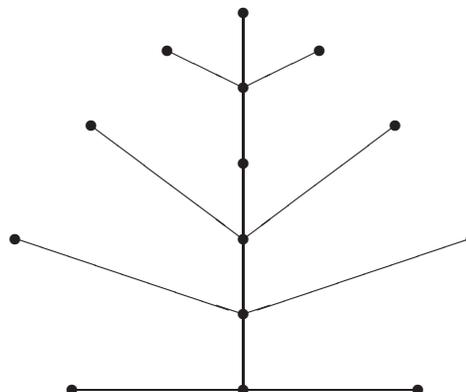


Рис. 4.

Для более равномерного распределения вершин по окружностям (Рис. 4) введем в рассмотрение понятие веса вершины. Обозначим корень v_0 . Пусть v – вершина графа G . Определим множество

$$T_v = \{u \in VG \mid d(v_0, u) > d(v_0, v) \text{ и } (v_0, u) \text{ – маршрут содержит } v\}.$$

Вес висячей вершины определим как ноль. Для вершины со степенью большей единицы вес определим следующей формулой

$$m(v) = \sum_{u \in T_v} m(u) + \frac{1 + (-1)^{m(u)}}{2}.$$

Геометрическая интерпретация формулы следующая: если степень вершины нечетная, то вершину можно расположить на радиусе векторе среднего потомка; когда же степень вершины четная, то она будет располагаться на отдельном радиусе векторе.

ЛИТЕРАТУРА

- [1] *Х. Томас* Часть VI. Алгоритмы для работы с графами. – М.: Вильямс, 2006. – 1296 с.
- [2] *G.Di. Battista, P. Eades, L.G. Tollis.* Algorithms for Drawings Graph: an Annotated Bibliography // Computational Geometry: Theory and Applications, 1994, no.4. – P. 235-282.

Реализация естественного отбора в генетическом алгоритме

Муфтиев Р. Р.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500И)

Эволюционные алгоритмы (evolutionary algorithms) [1, 2] широко применяются во многих областях искусственного интеллекта, оптимизации и других, сводящихся к поиску экстремумов функций одной или многих переменных. Они используют единую методологию на основе эволюционных принципов и генетических свойств наследственности для решения разнообразных задач моделирования и оптимизации. Сочетание «наследственность + изменчивость + естественный отбор» представляет пример «природной» реализации известного метода проб и ошибок, а потому претендует на универсальность применения наряду с последовательным перебором и случайным поиском.

Отбор «особей» для проведения операции «скрещивания» реализуется поразному. Как правило, он основан на некотором показателе приспособленности (fitness), вычисляемом для каждой особи на основе сравнительной близости к искомому решению. Однако для большей эффективности алгоритма отбор можно сделать гибким, внося некоторые параметры.

Пусть имеется популяция из n особей, для каждой из которых известно значение функции приспособленности F_i (i – порядковый номер особи). Необходимо путём естественного отбора сформировать следующую популяцию, тоже из n особей, с учётом того, что вероятности отбора должны быть пропорциональны значениям функции приспособленности. Это можно реализовать, размножив каждую особь в количестве экземпляров n_i , пропорциональном значению её приспособленности, и поместив все размноженные особи в один массив, из которого в результате и выбирается случайным равновероятным образом новые n особей. При этом вероятность выбора любой особи будет равна $p_i = \frac{n_i}{N}$, где N – размер всего массива. Наименьшее значение из всех n_i (n_{min}) будет представлено в массиве один раз (при этом на вероятность выбора этой особи будет влиять размер массива N). Таким образом, требуется подобрать отображение множества оценок приспособленности F_i на множество количеств n_i . Чтобы n_i было пропорционально F_i , зададим его как линейную функцию:

$$n_i = a * F_i + b \quad (1)$$

Коэффициенты a и b определим из дополнительных условий. Первое условие – это $n_{min} = 1$, а второе – сумма всех n_i должна равняться размеру массива.

Получается система двух линейных уравнений с двумя неизвестными:

$$\begin{cases} 1 & = a * F_{min} + b \\ N & = a * \sum_{i=1}^n F_i + n * b \end{cases} \quad (2)$$

Отсюда

$$a = \frac{N - n}{-(\sum_{i=1}^n F_i) + n * F_{min}} \quad b = \frac{-(\sum_{i=1}^n F_i) + F_{min} * N}{-(\sum_{i=1}^n F_i) + n * F_{min}} \quad (3)$$

Таким образом, получается алгоритм с параметром — размером массива N . Очевидно, чем больше N , тем меньше вероятность слабых особей выжить, а минимально возможное значение для N равно n — в этом случае $a=0$, $b=1$ и все $n_i = 1$, при этом вероятности выжить для всех особей популяции оказываются равными (т.е. собственно никакого отбора не происходит). Подобный контроль процесса отбора позволит настраивать его под определенный вид функции, снижая при этом время сходимости алгоритма.

ЛИТЕРАТУРА

- [1] Л.А. Гладков, В.М. Курейчик, В.В. Курейчик. *Генетические алгоритмы* — М.: Физматлит, 2006.
- [2] Т. Ваек, Д. Фогел, З. Михалевиц. *Evolutionary computation* — Berlin Heidelberg: Springer-Verlag, 2000.

Обобщенная формула Лейбница для (σ, δ)-производной

Налесная Е. П.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА АЛГЕБРЫ И ФУНКЦИОНАЛЬНОГО АНАЛИЗА (ГРУППА 500)

В данной статье выводится формула Лейбница для n -ой (σ, δ)-производной в банаховой алгебре \mathfrak{B} , которое является обобщением ранее полученных результатов М. Мирзавазиря и М.С. Мослехиана.

Пусть \mathfrak{B} -банахова алгебра, \mathfrak{A} -некоторая её подалгебра, $\sigma, \delta : \mathfrak{A} \rightarrow \mathfrak{B}$ -линейные отображения.

Определение 19. *Отображение d называется σ -производной, если $\forall a, b \in \mathfrak{A}$ выполняется соотношение*

$$d(ab) = d(a)\sigma(b) + \sigma(a)d(b).$$

σ -производные изучались различными авторами (см., например, [4]). В частности в [4] была получена формула Лейбница для n -ой σ -производной. Пусть теперь $\sigma, \delta : \mathfrak{A} \rightarrow \mathfrak{B}$ -отображения (линейность не обязательна).

Определение 20. *Линейное отображение $d : \mathfrak{A} \rightarrow \mathfrak{B}$ является (σ, δ)-производной [5], если $\forall a, b \in \mathfrak{A}$*

$$d(ab) = d(a)\sigma(b) + \delta(a)d(b)$$

Рассмотрим два семейства отображений $\{\varphi_{n,k}\}_{n \in \mathbb{N}, 0 \leq k \leq 2^n - 1}$ и $\{\psi_{n,k}\}_{n \in \mathbb{N}, 0 \leq k \leq 2^n - 1}$, которое определяется с помощью двоичного разложения числа $2^n - 1 = \underbrace{(1 \dots 1)}_n$. Определим отображение $\varphi_{n,k}$, следующим

образом. Если $k = (k_1 k_2 \dots k_s)_2$ -двоичное разложение k , то $\varphi_{n,k} = \phi_{k_1} \phi_{k_2} \dots \phi_{k_s}$, где $\phi_{k_i} = \begin{cases} d, & k_i = 1, \\ \sigma, & k_i = 0. \end{cases}$ Аналогично определяются отображения $\{\psi_{n,k}\}$. Если $k = (k_1 k_2 \dots k_s)_2$ -двоичное разложение k , то $\psi_{n,k} = \Psi_{k_1} \Psi_{k_2} \dots \Psi_{k_s}$, где $\Psi_{k_i} = \begin{cases} d, & k_i = 1, \\ \delta, & k_i = 0. \end{cases}$ Например, если $n = 5, k = 11 = (01011)_2$, то

$$\varphi_{5,11} = \sigma d \sigma d d = \sigma d \sigma d^2, \psi_{5,11} = \delta d \delta d d = \delta d \delta d^2.$$

Основные свойства отображений $\varphi_{n,k}$ и $\{\psi_{n,k}\}$ содержит

Лемма 4. *Пусть n -натуральное число и $0 \leq k \leq 2^n - 1$, тогда*

$$\begin{aligned}
d\psi_{n,k} &= \psi_{n+1,2^n+k} \\
d\varphi_{n,2^n-1-k} &= \varphi_{n+1,2^n-1-k} \\
\delta\psi_{n,k} &= \psi_{n+1,k} \\
\sigma\varphi_{n,2^n-1-k} &= \varphi_{n+1,2^{n+1}-1-(2^n+k)}
\end{aligned}$$

Имеет место

Теорема 10. Для любых $a, b \in \mathfrak{A}$ справедлива формула Лейбница

$$d^n(ab) = \sum_{k=0}^{2^n-1} \psi_{n,k}(a)\varphi_{n,2^n-1-k}(b) \quad (1)$$

Доверення. Докажем утверждение при помощи индукции по n . Для $n = 1$ имеем

$$d(ab) = d(a)\sigma(b) + \delta(a)d(b) = \psi_{1,1}(a)\varphi_{1,0}(b) + \psi_{1,0}(a)\varphi_{1,1}(b)$$

Теперь допустим, что (1) справедлива для любых натуральных $l \leq n$. Согласно лемме 4

$$\begin{aligned}
d^{n+1}(ab) &= d(d^n(ab)) = d\left(\sum_{k=0}^{2^n-1} \psi_{n,k}(a)\varphi_{n,2^n-1-k}(b)\right) = \\
&= \sum_{k=0}^{2^n-1} d(\psi_{n,k}(a)\varphi_{n,2^n-1-k}(b)) = \\
&= \sum_{k=0}^{2^n-1} (d(\psi_{n,k}(a))\sigma(\varphi_{n,2^n-1-k}(b)) + \delta(\psi_{n,k}(a))d(\varphi_{n,2^n-1-k}(b))) = \\
&= \sum_{k=0}^{2^n-1} \psi_{n+1,2^n+k}(a)\varphi_{n+1,2^{n+1}-1-(2^n+k)}(b) + \sum_{k=0}^{2^n-1} \psi_{n+1,k}(a)\varphi_{n+1,2^{n+1}-1-k}(b) = \\
&= \sum_{l=2^n}^{2^{n+1}-1} \psi_{n+1,l}(a)\varphi_{n+1,2^{n+1}-1-l}(b) + \sum_{l=0}^{2^n-1} \psi_{n+1,l}(a)\varphi_{n+1,2^{n+1}-1-l}(b) = \\
&= \sum_{l=0}^{2^{n+1}-1} \psi_{n+1,l}(a)\varphi_{n+1,2^{n+1}-1-l}(b)
\end{aligned}$$

□

Рассмотрим следующий пример. Пусть $n = 3$. Параметр k удовлетворяет двойному неравенству $0 \leq k \leq 7$. При этом

$$k = 0 = 0_2$$

$$k = 1 = 1_2$$

$$k = 2 = 10_2$$

$$k = 3 = 11_2$$

$$k = 4 = 100_2$$

$$k = 5 = 101_2$$

$$k = 6 = 100_2$$

$$k = 7 = 111_2$$

Соответствующие отображения $\varphi_{n,k}$ факторизуются следующим образом:

$$\varphi_{3,0} = \sigma\sigma\sigma = \sigma^3$$

$$\varphi_{3,1} = \sigma\sigma d = \sigma^2 d$$

$$\varphi_{3,2} = \sigma d\sigma = \sigma d\sigma$$

$$\varphi_{3,3} = \sigma d d = \sigma d^2$$

$$\varphi_{3,4} = d\sigma\sigma = d\sigma^2$$

$$\varphi_{3,5} = d\sigma d = d\sigma d$$

$$\varphi_{3,6} = d d\sigma = d^2\sigma$$

$$\varphi_{3,7} = d d d = d^3$$

Аналогичные равенства имеют место для отображений $\psi_{n,k}$:

$$\psi_{3,0} = \delta\delta\delta = \delta^3$$

$$\psi_{3,1} = \delta\delta d = \delta^2 d$$

$$\psi_{3,2} = \delta d\delta = \delta d\delta$$

$$\psi_{3,3} = \delta d d = \delta d^2$$

$$\psi_{3,4} = d\delta\delta = d\delta^2$$

$$\psi_{3,5} = d\delta d = d\delta d$$

$$\psi_{3,6} = dd\delta = d^2\delta$$

$$\psi_{3,7} = ddd = d^3$$

Формула Лейбница имеет вид:

$$\begin{aligned} d^3(ab) &= \psi_{3,0}(a)\varphi_{3,7}(b) + \psi_{3,1}(a)\varphi_{3,6}(b) + \\ &+ \psi_{3,2}(a)\varphi_{3,5}(b) + \psi_{3,3}(a)\varphi_{3,4}(b) + \psi_{3,4}(a)\varphi_{3,3}(b) + \\ &+ \psi_{3,5}(a)\varphi_{3,2}(b) + \psi_{3,6}(a)\varphi_{3,1}(b) + \psi_{3,7}(a)\varphi_{3,0}(b) = \\ &= \delta^3(a)d^3(b) + \delta^2 d(a)d^2\sigma(b) + \delta d\delta(a)d\sigma d(b) + \\ &+ \delta d^2(a)d\sigma^2(b) + d\delta^2(a)\sigma d^2(b) + d\delta d(a)\sigma d\sigma(b) + \\ &+ d^2\delta(a)\sigma^2 d(b) + d^3(a)\sigma^3(b). \end{aligned}$$

ЛИТЕРАТУРА

- [1] Gh. Abbaspour, M.s.Moslehian, A.Niknam. *Generalized Derivations on Modules.* //arXiv:Math.FA/0503618v1 26 Mar 2005.
- [2] M. Mirzavaziri, M.S. Moslehian. *A Generalization of the Kadison-Sakai theorem.* //arXiv:Math/0510267v1 [math.FA] 13 Oct 2005.
- [3] M. Mirzavaziri, M.S. Moslehian. *Automatic continuity of σ -derivations on C^* -algebras.* //arXiv:Math/0508028v2 [math.FA] 22 Apr 2006.
- [4] M. Mirzavaziri, M.S. Moslehian. *σ -derivations in Banach algebras.* //arXiv:Math/0505319v3 [math.FA] 22 Nov 2006.
- [5] S. Hejazian, A.R. Janfada, M. Mirzavaziri, M.S.Moslehian. *Achievement of continuity of (φ, ψ) -derivations without linearity.* //arXiv:Math/06116016v2[math.FA] 21 Jan 2007.

Малые колебания систем “идеальная жидкость-баротропный газ”

Насонкина А. В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА МАТЕМАТИЧЕСКОГО АНАЛИЗА (ГРУППА 500)

В данной работе рассматривается задача о малых колебаниях системы "идеальная жидкость – баротропный газ". Математическая постановка задачи формулируется для сосуда произвольной формы, полностью заполненного газом и жидкостью, а далее изучается случай, когда сосуд имеет цилиндрическую форму с произвольным сечением. Такая форма сосуда позволяет решать задачу методом разделения переменных. В результате получены два трансцендентных уравнения, которые решаются графическим способом. Одно уравнение имеет счетное множество решений $\{\lambda_{kl}\}_{k,l=1}^{\infty}$ с предельной точкой на бесконечности (λ – квадрат частоты колебаний); второе уравнение не имеет решений.

Будем считать, что идеальная несжимаемая жидкость с плотностью ρ_1 и баротропный газ, плотность которого ρ_2 , полностью заполняют неподвижный сосуд произвольной формы. Жидкость занимает в состоянии покоя область Ω_1 , частично ограниченную твердой стенкой S_1 , а газ, соответственно, заполняет область пространства Ω_2 , ограниченную твердой стенкой S_2 , а также границей раздела Γ_t . Будем считать, что равновесная поверхность Γ нам известна. В ее окрестности введем криволинейную систему координат ξ_1, ξ_2, ξ_3 так, чтобы поверхность Γ имела уравнение $\xi_3 = 0$, а координатные линии ξ_3 были при $\xi_3 = 0$ направлены по нормали n и имели коэффициент Ламе $l_3 = 1$. Тогда движущаяся свободная поверхность Γ_t жидкости имеет уравнение

$$\xi_3 = \zeta(\xi_1, \xi_2, t),$$

где ξ_3 , с точностью до малых высшего порядка, представляет собой отклонение Γ_t от Γ по нормали, взятое с соответствующим знаком.

Движение жидкости и газа зададим полями скоростей $\vec{u}_1(\vec{x}, t)$, $\vec{u}_2(\vec{x}, t)$ ($\vec{x} = (x, y, z)$) соответственно. Как обычно в теории малых движений жидкости, величины u_1 , u_2 и ζ и их производные по x , y , z , t считаем малыми первого порядка.

Приведем на основании уравнений и граничных условий, полученных в [1], математическую постановку задачи о малых движениях гидросистемы "идеальная жидкость – баротропный газ". Она состоит в отыскании потенциалов скоростей $\Phi_1(\vec{x}, t)$, $\Phi_2(\vec{x}, t)$ ($\vec{u}_k(\vec{x}, t) = -\nabla\Phi_k(\vec{x}, t)$, $k = 1, 2$) и функции смещения $\zeta(\xi_1, \xi_2, t)$, зависящих от t по закону $\exp(i\omega t)$ (ω – частота колебаний)

и удовлетворяющих следующим уравнениям и граничным условиям:

$$\Delta\Phi_1 = 0 \quad (\text{в } \Omega_1); \quad (1)$$

$$\frac{\partial\Phi_1}{\partial n} = 0 \quad (\text{на } S_1); \quad (2)$$

$$-\Delta\Phi_2 = \lambda c^{-2}\Phi_2 \quad (\text{в } \Omega_2); \quad (3)$$

$$\frac{\partial\Phi_2}{\partial n} = 0 \quad (\text{на } S_2); \quad (4)$$

$$\frac{\partial\Phi_1}{\partial n} = \frac{\partial\Phi_2}{\partial n} := \zeta \quad (\text{на } \Gamma); \quad (5)$$

$$\int_{\Gamma} \zeta \, d\Gamma = 0; \quad (6)$$

$$B_{\sigma}\zeta = \lambda(\rho_1\Phi_1 - \rho_2P_{\Gamma}\Phi_2) \quad (\text{на } \Gamma); \quad (7)$$

$$\lambda \int_{\Omega} \Phi_2 \, d\Omega_2 = 0. \quad (8)$$

Здесь \vec{n} – вектор нормали, причем предполагаем, что на Γ он направлен из Ω_1 в Ω_2 , а на стенках сосуда S_1, S_2 во внешнюю область; c – скорость звука в газе; λ – квадрат частоты колебаний: $\lambda = \omega^2$. Учитывая уравнение неразрывности [1], приходим к выводу, что Φ_1 гармоническая функция, т.е. имеет место (1). Условие (3) получается, если линеаризованное уравнение неразрывности [1] использовать для баротропного газа. Далее, (2), (4) – условия непротекания на смоченной части твердых стенок S_1 и S_2 , условие (5) – так называемое кинематическое условие, которое означает, что $u_{1n}|_{\Gamma}$ равно $u_{2n}|_{\Gamma}$ и это значение совпадает со скоростью перемещения вдоль нормали свободной движущейся поверхности Γ_t . Заметим, что в силу сохранения объема жидкости и газа, выполняется (6). Наконец, (7) – динамическое условие, которое получается из того факта, что разность давлений жидкости и газа равна капиллярному скачку. В этом условии B_{σ} – оператор, действующий следующим образом:

$$B_{\sigma}\zeta := P_{\Gamma}(-\sigma\Delta_{\Gamma}\zeta + a(\xi)\zeta), \quad D(B_{\sigma}) = \left\{ u \in \mathcal{H} \cap C^2(\Gamma) : \frac{\partial u}{\partial \nu} + \mathcal{X}u = 0 \right\}$$

($a(\xi)$, \mathcal{X} описаны в [1], ν – внешняя нормаль к границе Γ , построенная в плоскости касательной к Γ), а

$$P_{\Gamma}u := \left\{ u - \frac{1}{|\Gamma|} \int_{\Gamma} u \, d\Gamma = u - \hat{u} \right\}$$

– оператор проектирования на подпространство коразмерности 1:

$$\mathcal{H} = L_2(\Gamma) \ominus \{1_{\Gamma}\}.$$

Далее в работе будем рассматривать частный случай описанной задачи, когда сосуд ограничен цилиндрической поверхностью с вертикальными образующими, а снизу и сверху – частями горизонтальной плоскости произвольной формы. Тогда равновесной поверхностью служит горизонтальная плоскость: примем ее за плоскость $z = 0$. Будем считать, что верхняя граница имеет уравнение $z = h_2$, а нижняя – уравнение $z = -h_1$ ($h_1, h_2 > 0$). Угол смачивания δ равен 90° . Силы тяжести действуют вдоль оси Oz . Области Ω_1, Ω_2 могут быть записаны в виде декартового произведения двух множеств: $\Omega_1 = \Gamma \times (-h_1, 0)$, $\Omega_2 = \Gamma \times (0, h_2)$.

Такая форма сосуда позволяет нам провести разделение переменных. Функции Φ_1, Φ_2 будем искать в виде:

$$\Phi_1(x, y, z) = X(x, y)Z_1(z); \quad (9)$$

$$\Phi_2(x, y, z) = X(x, y)Z_2(z). \quad (10)$$

Подставляя (9), (10) в (1)–(8), получим задачи для функций X, Z_1, Z_2 :

$$\Delta_2 X + \mu^2 X = 0, \quad \left. \frac{\partial X}{\partial n} \right|_{\partial\Gamma} = 0, \quad \int_{\Gamma} X dx dy = 0; \quad (11)$$

$$Z_1'' - \mu^2 Z_1 = 0, \quad Z_1'(-h_1) = 0; \quad (12)$$

$$Z_2'' - (\mu^2 - \lambda c^{-2}) Z_2 = 0, \quad Z_2'(h_2) = 0, \quad (13)$$

где μ^2 – постоянная, возникающая при использовании метода разделения переменных. Задача (11) – это известная задача математической физики, которая имеет нам дискретный спектр $\{\mu_k^2\}$ ($k = \overline{1, \infty}$) и счетное множество собственных функций $X_k(x, y)$ ($k = \overline{1, \infty}$). Решая (12), получим:

$$Z_1 = C_{1k} \operatorname{ch}[\mu_k(z + h_1)] \quad (k = \overline{1, \infty}). \quad (14)$$

Для решения (13) необходимо рассмотреть 2 случая:

1-й случай: $\lambda c^{-2} - \mu_k^2 > 0$. Следовательно,

$$Z_2 = C_{2k} \cos(\sqrt{\lambda c^{-2} - \mu_k^2}(z - h_2)) \quad (k = \overline{1, \infty}). \quad (15)$$

Наша дальнейшая задача – найти для каждого μ_k соответствующее значение λ . Для решений (14), (15) необходимо потребовать, чтобы выполнялись кинематическое и динамическое условия (5) и (7). Из (5) следует:

$$C_{1k} \mu_k \operatorname{sh}(\mu_k h_1) = C_{2k} \sqrt{\lambda c^{-2} - \mu_k^2} \sin(\sqrt{\lambda c^{-2} - \mu_k^2} h_2). \quad (16)$$

Из (7) с учетом соотношения $-\Delta_2 X_k = \mu_k^2 X_k$, будем иметь:

$$\beta_k C_{1k} \mu_k \operatorname{sh}(\mu_k h_1) = \lambda(\rho_1 C_{1k} \operatorname{ch}(\mu_k h_1) - \rho_2(0) C_{2k} \cos(\sqrt{\lambda c^{-2} - \mu_k^2} h_2)), \quad (17)$$

где $\beta_k := \sigma\mu_k^2 + (\rho_1 - \rho_2(0))g > 0$.

Относительно неизвестных C_{1k} , C_{2k} однородная система (16) – (17) имеет нетривиальное решение тогда и только тогда, когда определитель этой системы равен нулю:

$$\begin{aligned} & \mu_k \operatorname{sh}(\mu_k h_1) \lambda \rho_2(0) \cos(\sqrt{\lambda c^{-2} - \mu_k^2} h_2) + \\ & + \sqrt{\lambda c^{-2} - \mu_k^2} \sin(\sqrt{\lambda c^{-2} - \mu_k^2} h_2) \cdot [\beta_k \mu_k \operatorname{sh}(\mu_k h_1) - \lambda \rho_1 \operatorname{ch}(\mu_k h_1)] = 0. \end{aligned} \quad (18)$$

Так как $\operatorname{sh}(\mu_k h_1) \cdot \sin(\sqrt{\lambda c^{-2} - \mu_k^2} h_2) \neq 0$ (это проверяется методом от противного), то можно (18) разделить на это выражение. Получим следующее трансцендентное уравнение:

$$\frac{\mu_k \lambda \rho_2(0) \operatorname{ctg}(\sqrt{\lambda c^{-2} - \mu_k^2} h_2)}{\sqrt{\lambda c^{-2} - \mu_k^2}} = \lambda \rho_1 \operatorname{cth}(\mu_k h_1) - \beta_k \mu_k. \quad (19)$$

Сделаем замену $w = \sqrt{\lambda c^{-2} - \mu_k^2} h_2$, запишем (19) в виде

$$\frac{\operatorname{ctg} w}{w} = \frac{\rho_1 \operatorname{cth}(\mu_k h_1)}{\rho_2(0) \mu_k h_2} - \frac{\sigma\mu_k^2 + (\rho_1 - \rho_2(0))g}{c^2 \rho_2(0) (\frac{w^2}{h_2^2} + \mu_k^2) h_2}. \quad (20)$$

Рассмотрим теперь **2-й случай**: $\mu_k^2 - \lambda c^{-2} > 0$. В таком случае:

$$Z_2 = C_{3k} \operatorname{ch}(\sqrt{\mu_k^2 - \lambda c^{-2}} (z - h_2)). \quad (21)$$

Проделив действия, описанные в **1-ом случае**, получим следующее трансцендентное уравнение:

$$\frac{\operatorname{cth} w}{w} = \frac{\rho_1 \operatorname{cth}(\mu_k h_1)}{\rho_2(0) \mu_k h_2} - \frac{\sigma\mu_k^2 + (\rho_1 - \rho_2(0))g}{c^2 \rho_2(0) (\mu_k^2 - \frac{w^2}{h_2^2}) h_2}, \quad (22)$$

где $w = \sqrt{\mu_k^2 - \lambda c^{-2}} h_2$.

Решая (20) и (22) графическим методом и с помощью автоматизированного пакета Maple, приходим к следующим выводам: уравнение (20) для каждого μ_k имеет счетное множество решений $\{\lambda_{kl}\}_{l=1}^{\infty}$ с предельной точкой на бесконечности; уравнение (22) не имеет решений.

ЛИТЕРАТУРА

- [1] Мышкис А.Д., Бабский В.Г., Жуков М.Ю., Копачевский Н.Д. и др. *Методы решения задач гидромеханики для условий невесомости*. – Киев: Наук. думка, 1992. – 592 с.

Применение технологий BPEL для создания бизнес-приложений

Павленко Е. А.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ (ГРУППА 500-П)

Одно из главных направлений развития современных информационных систем масштаба предприятия связано с концепцией сервис-ориентированной архитектуры (SOA). [1]

Service-Oriented Architecture (SOA) - это прикладная архитектура, в которой все функции определены как независимые сервисы с вызываемыми интерфейсами. Обращение к этим сервисам в определенной последовательности позволяет реализовать бизнес-процесс. [2]

Business Process Execution Language (BPEL) - универсальный язык для описания алгоритма выполнения бизнес-процессов. Язык часто рассматривается как ключевая составляющая сервис-ориентированной архитектуры приложений. [3]

Рассмотрим применение BPEL при решении следующей задачи: агентство получает от клиента запрос на приобретение товара, обрабатывает запрос и заказывает товар на фирме. В качестве среды исполнения процесса используется продукт Active Endpoints ActiveBPEL. Поставленная задача имеет следующий сценарий:

- Оператор заполняет форму с данными заявки клиента;
- Введенные данные отправляются на сервер исполнения BPEL-процесса;
- BPEL-процесс создает документ с текстом заявки в системе электронного документооборота;
- После создания документа BPEL-процесс направляет ответственному лицу задачу на рассмотрение заявки.

Первым шагом к использованию BPEL является создание набора приложений или бизнес-компонент, оформленных в виде web-сервисов. Под сервисами понимается функция приложения, представленная в виде компоненты, которая может использоваться для построения бизнес-процесса. Сервис либо возвращает какую-либо информацию по запросу, либо меняет состояние приложения, например заносит информацию в базу данных. В случае web-сервисов этот компонент имеет программный интерфейс, описанный на WSDL (Web Service Definition Language). [4]

Для исполнения поставленной задачи необходимо создать WSDL-файл. WSDL-описание процесса будет содержать один интерфейс, в котором содержится единая функция. В качестве входного параметра функция принимает сообщение. Это сообщение, содержит элемент, который несет в себе данные о новой заявке: регистрационный номер заявки (строка); краткое описание (строка); полный текст заявки (строка); срок выполнения, до наступления которого следует разрешить запрос (дата/время). Для того чтобы BPEL-процесс смог обращаться к внешним сервисам, необходимо импортировать в проект WSDL-файлы каждого используемого сервиса: сервис работы с документами в системе документооборота и сервис взаимодействия с диспетчером задач. Описание сервисов можно добавлять как в виде локального WSDL-файла, так и с помощью указания URL на внешний источник в сети.

Первое действие процесса состоит из элемента Receive, оно ожидает запроса со стороны клиента. Вызов данного действия будет приводить к созданию нового экземпляра процесса. Также создадим два действия типа Invoke, поскольку вызываем удаленные сервисы. Перед каждым вызовом удаленного сервиса добавим по элементу Assign, которые будут инициализировать переменные с входными данными для каждой вызываемой операции. Присвоим каждому действию дружественное название и заключим их в элемент Sequence. Все помещенные в элемент Sequence действия выполняются последовательно.

Закончив описание процесса, создадим клиентскую часть - форму заполнения заявки, используя Microsoft InfoPath.

С помощью технологий BPEL, мы автоматизировали процесс с участием нескольких информационных систем: системы электронного документооборота, диспетчера задач и Microsoft InfoPath.

ЛИТЕРАТУРА

- [1] А. Колесов *Автоматизация бизнес-процессов с помощью BPEL*. – Журнал "Byte/Россия", 2005.
- [2] Н. Дубова *SOA: подходы к реализации*. – Открытые системы, 2006.
- [3] И. Кулешов *Интеграция и автоматизация*. – Москва, 2006.
- [4] А. Гайдай *На пути к BPEL*, 2005.

Разработка системы дистанционного образования (e-learning)

Потапенко С.В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

Основные понятия

Дистанционное образование – метод обучения, при котором от обучаемого не требуется физического присутствия в определённом месте в процессе обучения. Как правило, тем или иным способом (например по почте) учащемуся предоставляются учебные материалы - учебники, видео и аудиокассеты и/или компакт-диски. Следуя методическим указаниям, студент самостоятельно (или с удалённой поддержкой учебного заведения) изучает материалы, выполняет задания и отправляет их на проверку обратно.

С развитием информационных технологий развивается и *e-learning* – форма дистанционного образования, в которой доставка учебного материала студентам, отправка работ для проверки, другое общение между участниками образовательного процесса осуществляется посредством электронных средств коммуникации (локальные или глобальные сети, локальные компьютерные обучающие системы) [1].

Участники образовательного процесса – Субъекты, участвующие в организации и проведении образовательного процесса в ДО. В системе e-learning это:

- *Администраторы* – осуществляют управление системой, регистрируют-удаляют пользователей системы, определяют их статус и права.
- *Методисты* – занимаются разработкой и размещением в системе учебных материалов.
- *Преподаватели (тьюторы)* – непосредственно общаются со студентами, проверяют выполнение самостоятельных работ, оценивают результаты промежуточных и квалификационных тестов.
- *Студенты* – потребители образовательных услуг.

Для каждого из субъектов предусмотрен соответствующий уровень доступа к учебным материалам и настройкам системы.

Модуль представляет из себя "квант" учебного процесса. Один учебный модуль соответствует занятию в СДО. Модуль может включать в себя теоретический материал, примеры решения практических задач, практические задания, промежуточное тестирование. Необходима возможность повторного

использования модулей в разных курсах, так чтобы можно было составлять новые учебные курсы из существующих в системе модулей, дополняя их новыми [2].

Учебный курс представляет из себя:

- (1) Упорядоченную последовательность учебных модулей, с указанием лимита времени для прохождения каждого. При этом первый модуль должен содержать вводную информацию о курсе в целом (Цели и задачи курса, знания и умения, которые должен усвоить студент во время его выполнения; аудитория, на которую рассчитан курс, базовые знания, которыми должен обладать студент перед прохождением курса) тестирование для проверки базовых знаний студента.
- (2) Завершающее квалификационное тестирование для проверки усвоения материала курса в целом после его завершения.

Технически *e-learning* – решение в форме образовательного портала строится на основе ряда программных продуктов:

- Система управления контентом/учебным порталом;
- Средство разработки учебного контента (Authoring Tool);
- Система управления обучением (Learning Management System);
- Система Дистанционного Обучения (СДО);
- Система обмена информацией.

Особенности реализации

Для реализации системы ДО может быть выбрана система управления контентом (CMS) с открытым кодом (Content Management System «Joomla!») по нескольким причинам:

- Система распространяется бесплатно.
- Система предназначена специально для поддержки информационных порталов (новостные сайты, онлайн-журналы, блоги и онлайн-дневники, сборники тематических статей, различные базы знаний и т.д.), и обладает мощными возможностями для систематизации, хранения и представления неограниченного количества структурированного текста.
- Система с открытым кодом. Обладает хорошо документированным API (Application program interface – набор функций, которые разработчик может использовать для доступа к функциональности программной компоненты (программы, модули и т.д.)), что позволяет расширять ее функциональность собственными модулями и компонентами.

- Система обладает готовой гибкой встроенной системой разграничения прав пользователей и администраторов.
- Систему легко "на ходу" дополняется новыми модулями, что позволяет легко расширять и модернизировать ее во время работы.

В качестве средства разработки учебного контента могут использоваться:

- Общеизвестные средства, ориентированные на Web-технологии и не использующие дорогостоящих специальных средств;
- инструментальные средства, специально ориентированные на разработку сетевых курсов.

В первую группу входят сравнительно недорогие или свободно распространяемые программные продукты. К ним относятся, например, редакторы HTML текстов, графические редакторы, конверторы форматов данных, возможно также применение средств создания аудио- и видео-фрагментов. В этой технологии удастся минимизировать первоначальные финансовые затраты, квалифицированные пользователи могут модернизировать и адаптировать созданные сетевые курсы. Однако создание сетевых курсов характеризуется при этом повышенными затратами времени [3].

Вышеперечисленные средства возможно использовать для создания учебных модулей. Система для построения и учета учебных курсов выключается в SMS как компонент-расширение.

Система управления обучением реализует следующие функции:

- Учет слушателей, персонализация, разграничение прав доступа
- Управление процессом обучения, учет результатов обучения и тестирования
- Интеграция с механизмами синхронного и асинхронного общения
- Подготовка аналитической отчетности
- Интеграция с внешними информационными системами (например, с системой кадрового учета)

Система реализуется как дополнительный компонент-расширение для SMS, выводящий в определенном формате результаты запросов к БД, хранящей всю информацию о студентах, преподавателях и учебных курсах, зарегистрированных в системе. Система должна распознавать несколько видов пользователей (Администратор, Методист, Преподаватель, Студент) наделяя каждого из них соответствующими правами.

Система обмена информацией представляет из себя набор программных средств, позволяющих обмен информацией между всеми субъектами учебного процесса выполняет следующие функции:

- Своевременная доставка учебных материалов студенту
- Обеспечение синхронного (онлайн-консультации посредством веб-чатов, виртуальные классные комнаты-синхронное общение нескольких студентов с преподавателем) и асинхронного (форумы, электронная почта) общения участников образовательного процесса.

Система реализуется как совокупность модулей и компонентов расширения CMS. Многие из необходимых компонент для данной CMS уже существуют и практически не требуют доработки для включения в систему (например форумы, веб-чаты, системы учета пользователей форумов-чатов) и распространяются бесплатно.

ЛИТЕРАТУРА

- [1] *Абрамова М.А.* Дистанционное обучение как показатель развития культуры сферы образования // Интернет-журнал "Эйдос". 2000, 25 апреля. <http://www.eidos.ru/journal/2000/0425-02.htm>. – В надзаг: Центр дистанционного образования "Эйдос", e-mail: list@eidos.ru.
- [2] *Ибрагимов. И. М.* Информационные технологии и средства дистанционного обучения: учебное пособие для студ. вузов. – М.: Академия, 2005. – 336 с.
- [3] *Попов А.Ю.* Разработка системы дистанционного образования. // Интернет-страница «СДО». <http://www.popov-science.narod.ru/>

Сравнительная характеристика сайтов АСМ-программирования

Сенчук А.В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. Вернадского
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (500и)

ВВЕДЕНИЕ

В век всеобщей компьютеризации, когда программирование для многих стало не только профессией или хобби, но и стилем жизни, все большую популярность приобретают сайты так называемого АСМ-программирования. Изначально АСМ-олимпиады задумывались как способ поиска наиболее одаренных в программировании молодых специалистов и направлены они, в первую очередь, на выявление студентов, имеющих не только глубокую теоретическую подготовку по всему спектру математических дисциплин, связанных с теоретической и практической информатикой и прикладной математикой, но и в совершенстве владеющих современными средствами разработки программного обеспечения, то есть способных применять теоретические знания в практическом эффективном программировании. Именно поэтому за право быть спонсором АСМ-олимпиады (и, конечно, приглашать к себе на работу участников олимпиады) ведется настоящая борьба среди ведущих компаний мира - производителей вычислительной техники и программного обеспечения. Участники финалов получают приглашения на работу в различных отделениях этих компаний после завершения учебы в университете. Сейчас АСМ-олимпиады - это одна из наиболее распространенных форм проведения международных студенческих соревнований по программированию, проводятся они при спонсорской и организационной поддержке Ассоциации компьютерной техники и всемирно известной компании IBM. [1]

Украинские команды принимают участие в АСМ-олимпиадах с 1995 года. В настоящее время Украина уверенно занимает лидирующие позиции в регионе, который объединяет страны юго-востока Европы.

Мощным средством для подготовки к участию в АСМ-олимпиадах являются сайты так называемого АСМ-программирования. Однако сайты АСМ-программирования предназначены не только для этого, но и для подготовки квалифицированных специалистов в области IT-технологий в целом, для тренировки в решении задач по программированию. На сайтах АСМ-программирования содержатся достаточно большие архивы задач различных

степеней сложности, поэтому желающие могут испытать свои силы в их решении, к тому же пользователям сайта дается возможность поучаствовать в периодически проводимых различных онлайн-соревнованиях по программированию. Такие возможности значительно повышают уровень знаний и умений молодых программистов, позволяют оценить реальный уровень собственной подготовки, приобрести опыт решения задач на время. В данной статье приводится обзор нескольких наиболее распространенных сайтов АСМ-программирования и приводится их сравнительная характеристика.

СРАВНИТЕЛЬНАЯ ХАРАКТЕРИСТИКА САЙТОВ АСМ-ПРОГРАММИРОВАНИЯ

Анализ большинства сайтов АСМ-программирования показал, что во всех подобных проектах присутствуют стандартные разделы:

- регистрация и изменение информации о пользователе;
- архив задач;
- система онлайн-проверки решения задач;
- система проведения онлайн-соревнований;
- форум;
- раздел, в котором приводятся ответы на наиболее часто задаваемые вопросы(FAQ);
- рейтинг участников;
- информация об авторах и о сайте.

Сайт Уральского государственного университета (<http://acm.timus.ru/>) [2] имеет очень удобный и понятный интерфейс. Хочется отметить возможность просмотра информации не только на английском, но и на русском языке. Приветственная страничка вполне оправданно называет сайт "сайтом, имеющим самый большой архив задач по программированию в России". Архив задач(problem set) действительно обширен, разделен на шесть групп и включает в общей сложности 1529 задач. Постоянные пользователи сайта отмечают, что задачи данного сайта отличаются нестандартностью, каждая задача особенная. Архив обновляется не очень часто - 3-4 раза в год. Предоставляется возможность отослать решения задач на онлайн-проверку, используя один из наиболее популярных в настоящее время языков программирования - C, ++, Pascal, Java или C#. Существует также вариант отправки решения на электронный адрес. Недостатком сайта является невозможность просмотра исходного кода отосланной на проверку задачи. На сайте регулярно проводятся онлайн-олимпиады, о проведении которых своевременно уведомляет специальный раздел проекта. Ведется рейтинг участников проекта. Еще

одной характерной особенностью данного проекта является перечень ссылок на некоторые сайты АСМ-программирования. Посетители активно используют возможности форума, на котором делятся опытом, помогают друг другу в решении затруднительных вопросов.

Польский сайт(<https://www.spoj.pl/>), дизайн которого отличается простотой, лаконичностью и удобством в использовании, обладает всеми необходимыми возможностями сайтов АСМ-программирования. Вся информация на сайте предоставлена на английском языке. Отличительной особенностью данного сайта является возможность отсылать задачи на множестве различных языков программирования(C, C++, Pascal, Java, PHP, Perl, Common Lisp, Fortran, Assembler, Nice, Smalltalk, ADA, Ruby, Bush и т.д.). На сайте ведется рейтинг участников онлайн-олимпиад, рейтинг по странам и рейтинг участников, регулярно отсылающих решения задач из архива на проверку. Задачи, представленные в архиве, характеризуются многими пользователями сайта как интересные и необычные. Форум сайта используется не слишком активно. [3]

Сайт Саратовского национального университета(<http://acm.sgu.ru>), команда программистов которого стала Чемпионом мира по программированию 2006 года, проигрывает вышеперечисленным сайтам в плане дизайнерского решения, интерфейс к сожалению не является достаточно понятным, имеется возможность просмотра только на английском языке. Сравнительно большой архив задач (всего 367 задач) предназначен скорее для опытных программистов, потому что представленные здесь задачи имеют достаточно высокий уровень сложности. На сайте реализованы функции отправки решений задач на распространенных языках программирования (C,C++, Delphi 6 и Java 1.5, Visual studio 7.1 C и Visual studio C++), создания виртуальные контексты(однако только на основе уже созданных), работает форум. [4]

Китайский сайт, найти который можно по адресу в Интернете <http://acm.tju.edu.cn/toj/> [5], выделяется из общего ряда сайтов с проверяющей системой возможностью очень удобного проведения виртуального конкурса для каждого отдельного пользователя - задачи можно выбрать непосредственно из архива задач. Архив задач в основном содержит задачи других сайтов АСМ-программирования, уникальных задач здесь практически не представлено. Быстрая регистрация, советы по использованию сайта в разделе FAQ делают работу легкой и доступной. Однако форум ведется на китайском языке, что затрудняет его использование пользователям из других стран.

ЗАКЛЮЧЕНИЕ

Сайты АСМ-программирования являются чрезвычайно важным и удобным инструментом не только для студентов, но и для преподавателей учебных учреждений, использующих возможности сайтов для составления заданий для учащихся. Сайтов АСМ-программирования становится все больше с каждым годом, но универсального сайта не существует, наверное, до сих пор. Каждый АСМ-сайт имеет свои слабые и сильные стороны, характерные особенности. Неоспоримым фактом является востребованность качественных сайтов АСМ-программирования и необходимость их создания и продвижения в сети Интернет.

ЛИТЕРАТУРА

- [1] <http://acm.uva.es/>
- [2] <http://acm.timus.ru/>
- [3] <https://www.spoj.pl/>
- [4] <http://acm.sgu.ru>
- [5] <http://acm.tju.edu.cn/toj/>

Медико-педагогическая база данных учащихся учебно-реабилитационного центра для детей с пониженным зрением

Синяков А. А.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ(ГРУППА 500-и)

Предпосылками для разработки системы управления медико-педагогической базы данных учащихся учебно-реабилитационного центра для детей с пониженным зрением послужило несколько причин, главными среди которых являются:

- необходимость обеспечения оптимизации образовательного процесса;
- потребность в систематизации данных, на основе существующего и нарабатанного материала;
- необходимость использования компьютерной эргономики;
- обеспечение процесса обучения в дистанционной среде [1].

Реализация системы, отвечающей всем вышеперечисленным требованиям, осуществлялась в несколько этапов:

1. Проектирование и построение базы данных.
2. Создание полнофункциональной системы управления и контроля информации. Разработка web портала.
 - 2.1 Создание административной части.
 - 2.2 Разработка система защиты и уровня доступа к информации.
 - 2.3 Построение систем регистрации, логирования и аутентификации.
 - 2.4 Создание индивидуальных информативных блоков для каждой группы пользователей.
3. Создание основных элементов портала: форум, блок новостей, гостевая книга, галерея, электронная библиотека.

Проектирование базы данных

По смысловой нагрузке база разделена на несколько функциональных блоков:

- общие данные об учащемся (имя, фамилия, пол, возраст, сведения о семье, о форме обучения, посещаемые кружки и секции, данные об участии в олимпиадах и конкурсах);
- медицинские данные учеников (медицинская карта, лист здоровья, динамика наблюдения, где и когда проходил лечение, история болезни);

- сведения об учителях и коррекционных педагогах (ФИО, преподаваемые предметы, логин, пароль для входа в свой сегмент базы);
- психолого-педагогическая характеристика (развернутые сведения-характеристики классного руководителя, учителей-предметников, дефектологов, психологов об учащемся);
- индивидуальные нормы работы за компьютером (время работы за монитором, размер шрифтов, цветовая гамма, разрешение экрана, используемые школьником программы и технические коррекционные средства);
- административный блок данных (списки пользователей, группы доступа к информации, системные настройки).

Кроме того, база содержит информацию, обеспечивающую стабильную и полнофункциональную работу всего портала.

Система управления базой данных

В качестве системы, обеспечивающей все необходимы критерии для удобной работы, а так же для контроля, управления и изменения информации, был разработан веб-портал. Портал позволяет любой группе пользователей, будь-то учитель, врач, ученик или гость получить доступ только к своему сегменту информации. Благодаря этому достигается удобство в работе с ресурсами сайта, а также надежность и конфиденциальность данных. Администраторы группы получают весь необходимый доступ к информации с возможностью ее просмотра и редактирования.

При входе в портал под своей учетной записью ученик попадает в индивидуальную среду, оптимально настроенную под его нужды. Настройки задаются на основании основного и сопутствующих диагнозов, а так же возраста и потребностей ученика.

Для учителей в портале предусмотрен индивидуальный раздел, представляющий ряд функциональных возможностей. Во-первых, это управление индивидуальной информацией (ФИО, преподаваемые предметы, логин, пароль для доступа к порталу). Во-вторых, каждый учитель имеет возможность управлять информацией об ученике, относящейся к его предметной области. Для составления психолого-педагогической характеристики учащегося, организован дневник наблюдений. Кроме этого для всех учителей есть раздел позволяющий в удобном виде, найти и просмотреть информацию о любом ученике.

В портале сформирован функциональный раздел, обеспечивающий полный доступ к медицинским данным учащихся. Для удобства работы, вся медицинская информация разделена на смысловые разделы: данные медицинской

карты; информация из листа здоровья; динамика наблюдения; где и когда проходил лечение; история болезни; офтальмо-эргономические данные.

В дальнейшем планируется доработка и усовершенствование web-портала школы, а также добавления ряда полезной функциональности:

- доработка блока, содержащего актуальную информацию о жизни школы (новости, доска объявлений и т. п.);
- создание электронной библиотеки, содержащей учебную, методическую и медицинскую литературу для учителей, родителей и школьников;
- создание электронных тетрадей и тестовых заданий с функциями оценивания и занесения результатов в базу данных школы;
- разработка отдела, содержащего коррекционные программы для развития зрения, внимания, восприятия.

Результатом работы является система, обеспечивающая полный контроль и управление базами данных, позволяющая проводить психолого-логопедо-медико-педагогического мониторинг состояния здоровья, физического, психического и умственного развития школьников. Данная система обладает широким набором возможностей, необходимых для полноценной реализации процесса обучения, в том числе, в дистанционной среде.

ЛІТЕРАТУРА

- [1] Косова Е. А. *Организация системы дистанционного образования для детей с пониженным зрением.* // Матеріали 5-ї міжнародної науково-технічної конференції "Проблеми підготовки та перепідготовки фахівців у сфері інформаційних технологій" (18-21 вересня, м.Севастополь) 2007г. с. 49-50. 0,5 п.л

Сравнение компактного субдифференциала с субдифференциалами Кларка и Мишеля-Пено

Столякин Ф. С.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
КАФЕДРА АЛГЕБРЫ И ФУНКЦИОНАЛЬНОГО АНАЛИЗА (ГРУППА 401)

Проводится сравнение недавно предложенного понятия компактного субдифференциала с субдифференциалами Кларка и Мишеля-Пено для вещественных функций.

Наличие множества существенных свойств субдифференциала выпуклых функций привело к различным аналогам и обобщениям этого понятия на невыпуклый случай (см., например [1]). В работах [2], [3] было введено новое в нелинейном анализе понятие *компактного субдифференциала* (*K-субдифференциала*) для отображений в локально выпуклые пространства (ЛВП) и исследованы его свойства. В этой заметке мы рассмотрим связь этого понятия с некоторыми другими современными типами субдифференциалов для вещественных функций $f : \mathbb{R} \rightarrow \mathbb{R}$.

В 1973 г. Ф. Кларк (см. [4]), а в 1984 г. Ф. Мишель и Ж.-П. Пено (см. [5]) предложили обобщения понятия производной по направлению. Пусть $x_0 \in \mathbb{R}$ — фиксированная точка, а $g \in \mathbb{R}$ — фиксированное направление. Положим

$$f_{Cl}^\uparrow(x_0; g) = \limsup_{x' \rightarrow x_0, \alpha \downarrow 0} \frac{1}{\alpha} [f(x' + \alpha g) - f(x')] ;$$

$$f_{mp}^\uparrow(x_0; g) = \sup_{q \in \mathbb{R}} \left\{ \limsup_{\alpha \downarrow 0} \frac{1}{\alpha} [f(x_0 + \alpha(g + q)) - f(x_0 + \alpha q)] \right\} .$$

Величины $f_{Cl}^\uparrow(x_0; g)$ и $f_{mp}^\uparrow(x_0; g)$ называются соответственно *верхними производными числами Кларка и Мишеля – Пено* в точке x_0 по направлению g . Обозначим через $\partial_{Cl}f(x_0)$ и $\partial_{mp}f(x_0)$ соответственно такие множества, что

$$f_{Cl}^\uparrow(x_0; g) = \max_{\ell \in \partial_{Cl}f(x_0)} (\ell \cdot g) ; \quad f_{mp}^\uparrow(x_0; g) = \max_{\ell \in \partial_{mp}f(x_0)} (\ell \cdot g) .$$

Если множество $\partial_{Cl}f(x_0)$ ($\partial_{mp}f(x_0)$) непусто и компактно, то оно называется *субдифференциалом Кларка (Мишеля – Пено)* функции f в точке x_0 .

Можно доказать, что компактная субдифференцируемость — более общее понятие, чем субдифференцируемость (наличие субдифференциала) по Кларку или Мишелю – Пено. При этом справедливы включения

$$\partial_K f(x_0) \subset \partial_{mp}f(x_0) \subset \partial_{Cl}f(x_0) ,$$

причём возможны ситуации

$$\partial_K f(x_0) = \partial_{mp} f(x_0) \subsetneq \partial_{Cl} f(x_0), \quad \partial_K f(x_0) \subsetneq \partial_{mp} f(x_0) = \partial_{Cl} f(x_0).$$

В частности отсюда следует, что часто используемое необходимое условие экстремума в терминах субдифференциалов Кларка и Мишеля – Пено (см. [1], [4]) является более слабым, чем соответствующее условие в терминах K -субдифференциалов ([2], [3]).

Выражаю признательность научному руководителю И.В.Орлову за внимание к работе и полезные обсуждения.

ЛИТЕРАТУРА

- [1] Демьянов В.Ф., Рощина В.А. Обобщённые субдифференциалы и экзостеры // Владикавказский математический журнал. — 2006. — Т. 8 — № 4. — С. 19-31.
- [2] Орлов И.В., Стонякин Ф.С. Компактные субдифференциалы: формула конечных приращений и смежные результаты // Современная математика. Фундаментальные направления — 18 с. — В печати.
- [3] Стонякин Ф.С. Компактный субдифференциал вещественных функций // Динамические системы — Симферополь: ТНУ, 2007. — Вып. 23 — С. 99 — 112.
- [4] Кларк Ф. Оптимизация и негладкий анализ. — М.: Наука, 1988. — 288 с.
- [5] Michel P., Penot J.P. Calculs sous-differential pour les fonctions lipschitziennes et non-lipschitziennes // C.R. Acad. Sc. Paris. Ser I. — 1984 — V. 298 — P. 269 — 272.

Исследование свойств интегро-дифференциального сплайна и его применение в геометрическом моделировании.

Таран О. Г.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ (ГРУППА 500-П)

Целью работы является исследование свойств интегро-дифференциального (ИД) сплайна. В основе работы лежит локальный многочлен Эрмита, применяемый как в численных методах, так и в теоретической математике. В стандартном случае этот многочлен определяется четырьмя константами, две из которых определяют крайние точки, еще две - значения касательных. Эти четыре "управляющие" константы формируют вид кривой на данном отрезке. В качестве весовых функций для этих четырех констант выбираются многочлены третьей степени, удовлетворяющие определенным свойствам.

Рассмотрим интегро-дифференциальный сплайн, включающий в себя условия равенства площади интерполируемой функции и сплайна на отрезке интерполирования $[x_i; x_{i+1}]$

$$S_i(x) = f_i\phi_1(x) + f_{i+1}\phi_2(x) + (f'_i\phi_3(x) + f'_{i+1}\phi_4(x))h_i + \frac{I_i^{i+1}\phi_5(x)}{h_i} \quad (1)$$

$$I_i^{i+1} = \int_{x_i}^{x_{i+1}} S(t)dt \quad (2)$$

Такой сплайн назовем интегро-дифференциальным (ИД сплайн). Физический смысл интеграла I_i^{i+1} - площадь под кривой, описываемой сплайном. Приравняв этот интеграл к интегралу от площади под аппроксимируемой кривой, добавляем к дифференциальным условиям интегральное условие согласования, то есть, накладываем дополнительное условие равенства площадей.

Очевидно, такой многочлен лучше приближает исследуемую функцию, так как он всегда будет иметь на участке интерполирования график, пересекающий график интерполируемой функции (рис 1).

При моделировании кривых и поверхностей чаще используются параметрические представления.

Пусть граница фигуры задана параметрическими уравнениями $x = x(t)$, $y = y(t)$. Для параметрических кривых существует несколько видов площадей: площадь сектора, площадь криволинейной трапеции, в проекции на ось абсцисс, площадь криволинейной трапеции, в проекции на ось ординат.

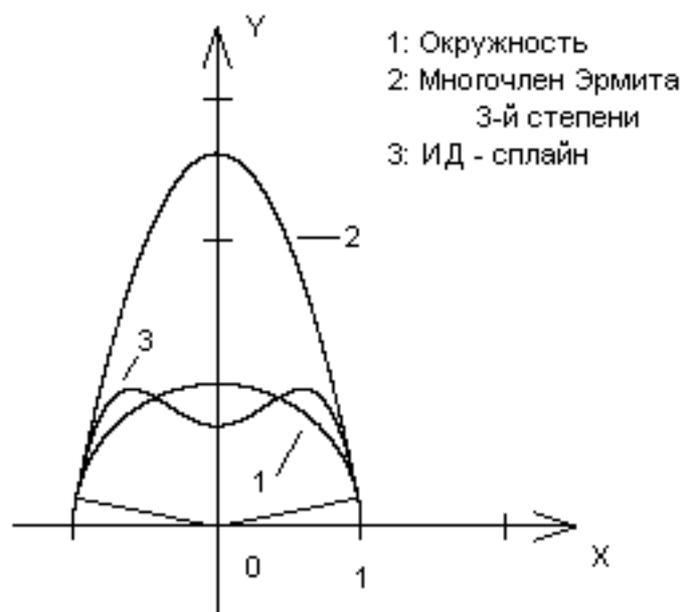


Рис. 1

В каждом из этих случаев определение базисных функций для ИД-сплайна приводит, естественно, к различному виду этих функций.

Рассмотрим несколько примеров различных площадей фигур, с использованием различных формул, показывающих существенные различия в результатах между стандартным кубическим сплайном и ИД-сплайном.

Формула площади сектора - рис 2, формула площади криволинейной трапеции, в проекции на ось абсцисс - рис 3.



Рис. 2

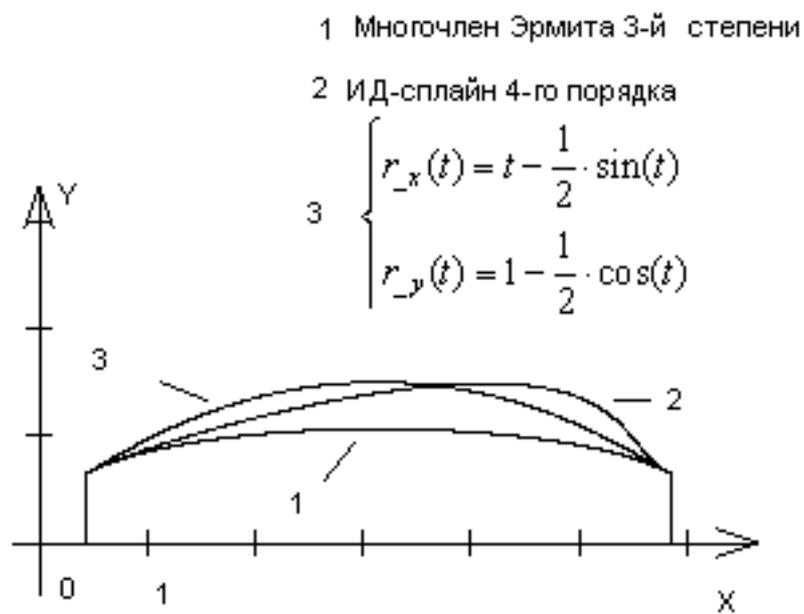


Рис. 3

При большом количестве параметров состояния построение системы высказываний «входы – выход» становится достаточно сложной задачей. Это обусловлено тем, то в один момент времени человек в состоянии оперировать не более, чем 7 ± 2 параметрами-признаками. В связи с этим уместно проводить иерархическую классификацию параметров состояния и по ней строить дерево вывода, которое будет определять систему вложенных высказываний меньшей размерности.

При иерархическом построении модели можно учесть практически неограниченное количество параметров состояний, влияющих на решение, а также упростить правила и уменьшить их количество в базе знаний.

Для математической постановки задачи введем следующие обозначения:

n – количество параметров, подаваемых на вход системы;

$x_i, i = \overline{1, n}$ – параметры, подаваемые на вход системы;

t_i – количество нечетких термов, используемых для оценки x_i ;

$y_i, i = \overline{1, m}$ – возможные варианты решения системы;

$b_i^j, j = \overline{1, t_i} \quad i = \overline{1, n}$ – координата центра функции принадлежности x_i к j -му терму;

$c_i^j, j = \overline{1, t_i} \quad i = \overline{1, n}$ – коэффициент концентрации функции принадлежности x_i к j -му терму;

$B = (b_1^1, \dots, b_1^{t_1}, b_2^1, \dots, b_2^{t_2}, \dots, b_n^1, \dots, b_n^{t_n})$ – вектор координат центров функций принадлежности,

$C = (c_1^1, \dots, c_1^{t_1}, c_2^1, \dots, c_2^{t_2}, \dots, c_n^1, \dots, c_n^{t_n})$ – вектор коэффициентов концентрации функций принадлежности,

M – размер обучающей выборки;

$(\hat{X}_l, \hat{y}_l), l = \overline{1, M}$, где $\hat{X}_l = (x_1^l, \dots, x_n^l), \hat{y}_l \in \{y_i | i = \overline{1, m}\}$ – нечеткая обучающая выборка.

$W = (w_1, \dots, w_M)$ – вектор весовых коэффициентов нечетких правил;

Сущностью этапа обучения модели является подбор таких весов нечетких правил базы знаний и таких параметров функции принадлежности, которые минимизируют отклонения между экспериментальными данными и результатами нечеткого вывода:

$$\sqrt{\frac{1}{M} \sum_{l=1, M} \left(q(\hat{X}^l, \hat{y}^l, B, W, C) \right)^2} \longrightarrow \min \quad (1)$$

где $q(\hat{X}^l, \hat{y}^l, B, W, C)$ – расхождение между экспериментальным выходом \hat{y}^l и результатом нечеткого логического вывода, заданного моделью (B, C, W) , для

объекта X^l . Расхождение может быть рассчитано по следующей формуле:

$$q(\hat{X}^l, \hat{y}^l, B, W, C) = \sum_{j=1}^m \left(\mu^{y_j}(\hat{y}^l) - \bigvee_{\substack{p=\overline{1, M} \\ \hat{y}_p = y_j}} \left(w_p \bigwedge_{i=1}^n \mu^{\hat{x}_i^p}(\hat{x}_i^l, B, C) \right) \right) \quad (2)$$

где $\mu^{y_j}(\hat{y}^l) = \begin{cases} 1, & y_j = \hat{y}^l \\ 0, & y_j \neq \hat{y}^l \end{cases}$ — степень принадлежности экспериментально определённого решения \hat{y}^l к нечетким термам y_j , $j = \overline{1, m}$.

Обучение данной модели сводится к задаче нелинейной оптимизации, поэтому для больших баз знаний уместно использовать генетические алгоритмы, которые позволяют находить глобальный оптимум в мультимодальных задачах оптимизации и, зачастую, значительно эффективней обычных методов математического программирования. Генетические алгоритмы являются аналогом случайного поиска, ведущегося одновременно из разных начальных точек. В этом случае потенциальные решения представляются в виде особей, т.е. векторов, координатами которых являются параметры модели. Поскольку изначально нет сведений о локации оптимума, для создания исходной популяции будем использовать стратегию «дробовика», т.е. генерацию достаточно большого случайного подмножества решений.

Отбор родителей производится путем выбора наиболее приспособленных индивидуумов. При этом отборе используют целевую функцию, как единственно доступный источник информации о качестве решения. Каждой особи ставится в соответствие значение функции соответствия, и чем больше это значение для некоторого индивидуума, тем больше вероятность того, что он даст потомство. Операция скрещивания или кроссинговер, заключается в обмене частями хромосом между парой особей. После чего особи популяции с некоторой вероятностью подвергаются мутации, которая предусматривает случайное изменение элементов хромосомы.

Необходимой процедурой в генетическом алгоритме является редукция — устранение «плохих» решений, полученных в процессе формирования новой популяции. Промежуточную популяцию составляют все особи t поколения и особи, полученные в результате скрещивания и мутации. Тогда численность этой популяции можно определить следующим образом:

$$R^{t+1} = r^t + r_{cr}^t + r_m^t,$$

где r^t — размер популяции на предыдущем шаге; r_{cr}^t — число особей, полученных путем скрещивания; r_m^t — число «мутантов». Для того, чтобы размер

популяции был постоянным в ходе работы алгоритма, будем использовать селекционную схему. В этом случае родители и потомки, выступая на равных правах, помещаются в одну репродуктивную группу, используя процедуру отбора, дающую приоритет лучшим решениям. Очевидно, что каждое следующее поколение будет в среднем лучше предыдущего. Когда приспособленность индивидуумов перестает заметно увеличиваться, процесс останавливают и в качестве решения задачи оптимизации берут наилучшего из найденных индивидуумов.

ЛІТЕРАТУРА

- [1] Панкевич О.Д., Штовба С.Д. Діагностування тріщин будівельних конструкцій за допомогою нечітких баз знань. Монографія. – Вінниця: УНІВЕРСУМ-Вінниця. 2005. – 108 с.
- [2] Ротштейн А.П. Интеллектуальные технологии идентификации: нечеткая логика, генетические алгоритмы, нейронные сети. – Винница: УНІВЕРСУМ-Вінниця. 1999. – 320 с.
- [3] Леоненков А.В. Нечеткое моделирование в среде MATLAB и fuzzyTECH. – СПб: БВХ-Петербург. 2003. – 736 с.: ил.
- [4] Штовба С.Д. Настройка нечеткой модели по обучающей выборке с нечетким выходом // Кибернетика и системный анализ. – 2007. – № 3. – С. 26-32.
- [5] Ротштейн А.П., Штовба С.Д. Идентификация нелинейной зависимости нечеткой базой знаний с нечеткой обучающей выборкой // Кибернетика и системный анализ. – 2006. – № 2. – С. 17-24.
- [6] Скобцов Ю.А., Скобцов В.Ю. Современные модификации и обобщения генетических алгоритмов // Таврический вестник информатики и математики. – 2004. – № 1. – С.60-71.

Применение графа нормальных слов для вычисления размерности и роста алгебры

Тупикова А.В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ (ГРУППА 500П)

Объектом изучения настоящей работы являются алгебры, заданные образующими и соотношениями. Исследуется возможность применения алгоритмического подхода [1] для вычисления *размерности и роста* рассматриваемых алгебр.

1. Определение алгебры

Определение 21. *Алгебра A – это линейное пространство над некоторым полем K , с введенным на нем законом умножения $*$, удовлетворяющим требованиям линейности и дистрибутивности. $(\alpha*a + \beta*b)*c = \alpha*a*c + \beta*b*c$, где $\alpha, \beta \in K; a, b, c \in A$.*

Рассмотрим некоторый алфавит $X = \{x_1, x_2, \dots, x_n\}$. Из букв будем составлять слова. Линейные комбинации слов являются элементами так называемой свободной алгебры, заданной образующими x_1, x_2, \dots, x_n . Обозначим $A = \mathbb{C}\langle x_1, x_2, \dots, x_n \rangle$ – свободная алгебра над полем комплексных чисел. Элементы алгебры A это некоммутативные многочлены.

Соотношения – это равенства вида:

$$\begin{aligned} f_1(x_1, x_2, \dots, x_n) &= 0 \\ f_2(x_1, x_2, \dots, x_n) &= 0 \\ &\dots \\ f_r(x_1, x_2, \dots, x_n) &= 0 \end{aligned}$$

Где f_1, f_2, \dots, f_r элементы алгебры A .

Обозначим **алгебру**, заданную образующими и соотношениями, следующим образом:

$$A = \mathbb{C}\langle x_1, x_2, \dots, x_n \mid f_1(x_1, x_2, \dots, x_n) = 0, \dots, f_r(x_1, x_2, \dots, x_n) = 0 \rangle.$$

Таким образом, алгебра, заданная образующими и соотношениями состоит из *некоммутативных* многочленов, часть из которых является нулевым элементом и следовательно, некоторые элементы равны между собой.

2. Базис Гребнера

Эффективным средством решения ряда проблем, связанных с алгебрами, заданными образующими и соотношениями, является использование базиса Гребнера. Рассмотрим идеал I - порожденный соотношениями f_1, \dots, f_r .

Определение 22. Подмножество G идеала I называется базисом Гребнера, если $\forall v \in I$ найдется элемент G старший член которого является подсловом старшего слова v .

Старший член многочлена определяется относительно однородного лексикографического порядка.

Теперь приведем две теоремы [2], поясняющие важность базиса Гребнера при исследовании алгебр, заданных образующими и соотношениями.

Теорема 11. Размерность алгебры A равна количеству нормальных слов

Теорема 12. Слово s является нормальным \Leftrightarrow ни одно старшее слово элементов базиса Гребнера не является подсловом s .

3. Граф нормальных слов

Введем следующие обозначения: если f, g – слова, то $f \triangleleft g$ означает, что f – собственное начало g , а $f \trianglelefteq g \Leftrightarrow (f \triangleleft g) \vee (f = g)$.

Обозначим через F множество обструкций (множество старших слов элементов базиса Гребнера). Основной прием вычисления размерности и роста алгебры дает следующая теорема:

Теорема 13. Пусть V – множество всех собственных концов обструкций, объединенное со множеством образующих X . Тогда существует взаимно однозначное соответствие между маршрутами в графе и множеством непустых нормальных слов.

Вычисление размерности алгебры, ассоциированной с диаграммой Дынкина A_3

диаграмма Дынкина A_3

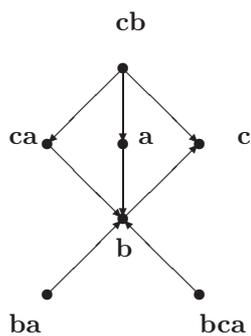
Алгебра, ассоциированная с данной диаграммой:

$$\begin{aligned}
 A_{A_3, \tau}^1 = \mathbb{C}\langle a, b, c \mid & aba - bab - \tau a + \tau b = 0 \\
 & bcb - cbc - \tau b + \tau c = 0 \\
 & ac - ca = 0 \\
 & a^2 - a = 0 \\
 & b^2 - b = 0 \\
 & c^2 - c = 0 \rangle
 \end{aligned}$$

Множество обструкций (множество старших слов Базиса Гребнера, соответствующего данной алгебре) $F = \{aba, bcb, ac, aa, bb, cc, abca\}$, $|F| = 7$.

Множество вершин графа нормальных слов выглядит следующим образом:
 $V = \{a, b, c, ba, cb, bca, ca\}$

Граф нормальных слов выглядит следующим образом:



Матрица смежности для данного графа:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \text{ Умно-}$$

жаем матрицу саму на себя до тех пор, пока она не станет нулевой. В результате получаем, что путей длины 0-7, путей длины 1-8, путей длины 2-6. Итого 23 и плюс 1 $\Rightarrow \dim A_{3,\tau} = 24$

В дипломной работе реализована программа построения графа нормальных слов и планируется изучить размерности и рост ряда конкретных алгебр.

ЛИТЕРАТУРА

- [1] Ostrovskiy V.L., Samoilenko Yu.S. Introduction to the theory representation of finitely presented *-algebras. 1 Representation by bounded operators. Rev. Math and Math. Phys., Gordon and Breach.-1999.-Vol.11.-P.1-261.
- [2] Уфнаровский В.А. Интеллектуальные современные проблемы математики. Фундаментальные направления. Т.57.1. Комбинаторные и асимптотические методы в алгебре.-М.: ВИНТИ, 1990.-272с.

Распознавание образов на основе нейронных сетей встречного распространения

Чернышёвой Ю.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

Образ, класс - классификационная группировка в системе классификации, объединяющая определенную группу объектов по некоторому признаку.

Образное восприятие мираж- одно из загадочных свойств живого мозга, позволяющее разобраться в бесконечном потоке воспринимаемой информации и разрозненных данных о внешнем мире. Воспринимая внешний мир, мы всегда производим классификацию воспринимаемых ощущений, т. е. разбиваем их на группы похожих, но не тождественных явлений. Образы обладают характерным свойством, проявляющимся в том, что ознакомление с конечным числом явлений из одного и того же множества дает возможность узнавать сколь угодно большое число его представителей. Образы обладают характерными объективными свойствами в том смысле, что разные люди, обучающиеся на различном материале наблюдений, большей частью одинаково и независимо друг от друга классифицируют одни и те же объекты. В связи с этим возникает проблема распознавания образов, которая состоит из двух частей: обучения и распознавания. Обучение осуществляется путем показа отдельных объектов с указанием их принадлежности тому или другому образу. В результате обучения распознающая система должна приобрести способность реагировать одинаковыми реакциями на все объекты одного образа и различными - на все объекты различных образов. Очень важно, что процесс обучения должен завершиться только путем показов конечного числа объектов без каких-либо других подсказок. В качестве объектов обучения могут быть либо картинки, либо другие визуальные изображения (буквы), либо различные явления внешнего мира, например звуки, состояния организма при медицинском диагнозе, состояние технического объекта в системах управления и другие. Важно, что в процессе обучения указываются только сами объекты и их принадлежность образу. За обучением следует процесс распознавания новых объектов, который характеризует действия уже обученной системы. Автоматизация этих процедур и составляет проблему обучения распознаванию образов.[1].

Основная трудность проблемы распознавания образов состоит в том, что на первый взгляд все методы и алгоритмы решения её кажутся совершенно

различными и часто никакой из них не годится для решения той задачи, которую необходимо решить. Очевидно, именно это привело к возникновению огромного множества алгоритмов, предназначенных для решения задачи распознавания образов. Один из методов основан на использовании нейронной сети. Работа сети разделяется на обучение и адаптацию. Под обучением понимается процесс адаптации сети к предъявляемым эталонным образцам путем модификации в соответствии с тем или иным алгоритмом весовых коэффициентов связей между нейронами.

Среди различных структур нейронных сетей одной из наиболее известных является многослойная структура, в которой каждый нейрон произвольного слоя связан со всеми весами нейронов предыдущего слоя. К многослойным сетям относятся сети встречного распространения.

[1]. Нейронная сеть встречного распространения обучается на выборке пар векторов (X, Y) в задаче представления отображения $X \rightarrow Y$. Замечательной особенностью этой сети является способность к обучению и отображению совокупности X, Y в себя. При этом благодаря обобщению, появляется возможность восстановления пары (X, Y) по одной известной компоненте (X или Y). При предъявлении на этапе распознавания только вектора X (с нулевым начальным Y) производится прямое отображение - восстанавливается Y , и наоборот, при известном Y может быть восстановлен соответствующий ему X . Возможность решения как прямой, так и обратной задачи, а также задачи по восстановлению отдельных недостающих компонент делает данную нейросетевую архитектуру уникальным инструментом.[1].

[2]. Во встречном распространении объединены два известных алгоритма: самоорганизующаяся карта Кохонена и звезда Гроссберга. При этом появляются свойства, которых нет ни у одного из них в отдельности. В процессе обучения входные векторы ассоциируются с соответствующими выходными векторами; они могут быть двоичными, состоящими из нулей и единиц, или непрерывными. Когда сеть обучена, приложение входного вектора приводит к требуемому выходному вектору. Обобщающая способность сети позволяет получать правильный выход даже при приложении входного вектора, который является неполным или слегка неверным. Таким образом, возможно использовать данную сеть для распознавания образов, восстановления образов и усиления сигналов. Как и многие другие сети, встречное распространение функционирует в двух режимах: в нормальном режиме, при котором принимается входной вектор X и выдается выходной вектор Y , и в режиме обучения, при котором подается входной вектор и веса корректируются, чтобы дать требуемый выходной вектор.

Нейроны слоя 0 (Приложение 1) служат лишь точками разветвления и не выполняют вычислений. Каждый нейрон слоя 0 соединен с каждым нейроном слоя 1 (называемого слоем Кохонена) отдельным весом W_{mn} . Эти веса в целом рассматриваются как матрица весов W . Аналогично, каждый нейрон в слое Кохонена (слое 1) соединен с каждым нейроном в слое Гроссберга (слое 2) весом V_{np} . Эти веса образуют матрицу весов V . Слой Кохонена классифицирует входные векторы в группы схожих. Это достигается с помощью такой подстройки весов слоя Кохонена, что близкие входные векторы активируют один и тот же нейрон данного слоя. Затем задачей слоя Гроссберга является получение требуемых выходов.[2].

Среди множества задач по распознаванию следует выделить задачу распознавания изображений, например, распознавание графических данных в формате .jpeg.

Предположим, перед нами стоит задача обучить нейронную сеть распознавать лица людей, то есть провести "идентификацию". Фотографии людей при этом представлены в формате .jpeg. Поставленная задача может быть решена с помощью применения сетей встречного распространения. Фотография представляет собой определённую последовательность бит. Следовательно, на вход нейронной сети необходимо подавать этот набор бит. Набор бит, в свою очередь, представляет собой вектор значений.[3]. Перед тем, как предъявлять сети входные векторы, их следует нормализовать. Операция выполняется с помощью деления каждой компоненты входного вектора на длину вектора. Эта длина находится извлечением квадратного корня из суммы квадратов компонент вектора. В алгебраической записи

$$S'_i = \frac{x_i}{\sqrt{x_1^2 + x_2^2 + \dots + x_n^2}}$$

Таким образом, входной вектор превращается в единичный вектор с тем же самым направлением, т.е. в вектор единичной длины в n -мерном пространстве.

После подачи данных на вход нейронной сети начинается процесс обучения. В своей простейшей форме слой Кохонена функционирует в духе "победитель забирает все", т.е. для данного входного вектора один и только один нейрон Кохонена выдает на выходе логическую единицу, а все остальные выдают ноль. Ассоциированное с нейронами Кохонена множество весов связывает каждый нейрон с каждым входом. Например, нейрон Кохонена K_1 имеет веса $W_{11}, W_{21}, \dots, W_{ml}$, составляющие весовой вектор W_1 . Они соединяются через входной слой с входными сигналами x_1, x_2, \dots, x_m , составляющими

входной вектор X . Подобно нейронам большинства сетей, выход \mathcal{NET} каждого нейрона Кохонена является просто суммой взвешенных входов. Это может быть выражено следующим образом:

$$\mathcal{NET}_j = \sum_i x_i W_{ij}$$

, где \mathcal{NET}_j - это выход \mathcal{NET} нейрона Кохонена j , или, в векторной записи,

$$\mathcal{N} = XW$$

, где \mathcal{N} - вектор выходов \mathcal{NET} слоя Кохонена. Нейрон Кохонена с максимальным значением \mathcal{NET} является "победителем". Его выход равен единице, у остальных он равен нулю.

Слой Кохонена классифицирует входные векторы в группы схожих. Это достигается с помощью такой подстройки весов слоя Кохонена, что близкие входные векторы активируют один и тот же нейрон данного слоя. Обучение Кохонена является самообучением, протекающим без учителя. При обучении слоя Кохонена на вход подается входной вектор и вычисляются его скалярные произведения с векторами весов, связанными со всеми нейронами Кохонена. Нейрон с максимальным значением скалярного произведения объявляется "победителем", и его веса подстраиваются. Так как скалярное произведение, используемое для вычисления величин \mathcal{NET} , является мерой сходства между входным вектором и вектором весов, то процесс обучения состоит в выборе нейрона Кохонена с весовым вектором, наиболее близким к входному вектору, и дальнейшем приближении весового вектора к входному. Сеть самоорганизуется таким образом, что данный нейрон Кохонена имеет максимальный выход для данного входного вектора. Уравнение, описывающее процесс обучения, имеет следующий вид:

$$W_n = W_s + \alpha(x - W_s)$$

, где W_n - новое значение веса, соединяющего входную компоненту x с выигравшим нейроном; W_s - предыдущее значение этого веса; α - коэффициент скорости обучения, который может варьироваться в процессе обучения.

Каждый вес, связанный с выигравшим нейроном Кохонена, изменяется пропорционально разности между его величиной и величиной входа, к которому он присоединен. Направление изменения минимизирует разность между весом и его входом.

Переменная α является коэффициентом скорости обучения, который вначале обычно равен 0.7 и может постепенно уменьшаться в процессе обучения. Это позволяет делать большие начальные шаги для быстрого грубого обучения и меньшие шаги при подходе к окончательной величине.

Всем весам сети перед началом обучения следует придать начальные значения. Общепринятой практикой при работе с нейронными сетями является присваивание весам небольших случайных значений. При обучении слоя Кохонена случайно выбранные весовые векторы следует нормализовать. Окончательные значения весовых векторов после обучения совпадают с нормализованными входными векторами. Поэтому нормализация перед началом обучения приближает весовые векторы к их окончательным значениям, сокращая, таким образом, продолжительность обучающего процесса.

Рандомизация весов слоя Кохонена может породить серьезные проблемы при обучении, так как в результате весовые векторы распределяются равномерно по поверхности гиперсферы. Из-за того, что входные векторы, как правило, распределены неравномерно и имеют тенденцию группироваться на относительно малой части поверхности гиперсферы, большинство весовых векторов будут так удалены от любого входного вектора, что они никогда не смогут дать наилучшее соответствие. Эти нейроны Кохонена будут всегда иметь нулевой выход и окажутся бесполезными. Более того, оставшихся весов, дающих наилучшие соответствия, может оказаться слишком мало, чтобы разделить входные векторы на классы, которые расположены близко друг к другу на поверхности гиперсферы.[3].

Допустим, что имеется несколько множеств входных векторов, все эти множества сходные, но необходимо разделить их на различные классы. Сеть должна быть обучена активировать отдельный нейрон Кохонена для каждого класса. Если начальная плотность весовых векторов в окрестности обучающих векторов слишком мала, то, возможно, не удастся разделить сходные классы из-за того, что весовых векторов в интересующей нас окрестности не хватит, чтобы приписать по одному из них каждому классу входных векторов.

Наоборот, если несколько входных векторов получены незначительными изменениями из одного и того же образца и должны быть объединены в один класс, то они должны включать один и тот же нейрон Кохонена. Если же плотность весовых векторов очень высока вблизи группы слегка различных входных векторов, то каждый входной вектор может активировать отдельный нейрон Кохонена. Наиболее желательное решение будет таким: распределить весовые векторы в соответствии с плотностью входных векторов, подлежащих разделению, и для этого поместить больше весовых векторов в окрестности большого числа входных векторов. Существует несколько методов приближенного достижения этой цели.

[1]. Одно из решений, известное под названием метода выпуклой комбинации состоит в том, что все веса приравниваются к одной и той же величине

$$W_i = \frac{1}{\sqrt{n}}$$

, где n - число входов и, следовательно, число компонент каждого весового вектора. Благодаря этому все весовые векторы совпадают и имеют единичную длину. Каждой же компоненте входа X придается значение

$$x_i = \alpha x_i + \frac{1 - \alpha}{\sqrt{n}}$$

, где n - число входов. В начале α очень мало, вследствие чего все входные векторы имеют длину, близкую к $\frac{1}{\sqrt{n}}$, и почти совпадают с векторами весов. В процессе обучения сети α постепенно возрастает, приближаясь к единице. Это позволяет разделять входные векторы и окончательно приписывать им их истинные значения. Весовые векторы отслеживают один или небольшую группу входных векторов и в конце обучения дают требуемую картину выходов. [1].

После того, как слой Кохонена окажется обученным, происходит обучение слоя Гроссберга. [3]. Входной вектор, являющийся выходом слоя Кохонена, подается на слой нейронов Гроссберга. Его выход \mathcal{NET} является взвешенной суммой выходов k_1, k_2, \dots, k_n слоя Кохонена, образующих вектор K . Вектор соединяющих весов, обозначенный через V , состоит из весов $V_{11}, V_{21}, \dots, V_{np}$. Тогда выход \mathcal{NET} каждого нейрона Гроссберга есть

$$\mathcal{NET}_j = \sum_i k_i V_{ij},$$

где \mathcal{NET}_j - выход j -го нейрона Гроссберга, или, в векторной форме,

$$\mathcal{N} = KV,$$

где \mathcal{N} - выходной вектор слоя Гроссберга, K - выходной вектор слоя Кохонена, V - матрица весов слоя Гроссберга.

Если слой Кохонена функционирует таким образом, что лишь у одного нейрона величина \mathcal{NET} равна единице, а у остальных равна нулю, то всего один элемент вектора K отличен от нуля и вычисления очень просты. Фактически каждый нейрон слоя Гроссберга лишь выдает величину веса, который связывает этот нейрон с единственным ненулевым нейроном Кохонена. Далее, каждый вес корректируется только в том случае, если он соединен с нейроном Кохонена, имеющим ненулевой выход. Величина коррекции веса пропорциональна разности между весом и требуемым выходом нейрона Гроссберга, с

которым этот вес соединен. В символьной записи

$$V_{ijn} = V_{ijs} + \beta(y_j - V_{ijs})k_i$$

, где k_i - выход i -го нейрона Кохонена, y_j - j -я компонента вектора желаемых выходов.

Первоначально β берется равным приблизительно 0,1 и затем постепенно уменьшается в процессе обучения. Веса слоя Гроссберга будут сходиться к средним величинам от желаемых выходов, тогда как веса слоя Кохонена обучаются на средних значениях входов. Обучение слоя Гроссберга - это обучение с учителем, алгоритм располагает желаемым выходом, по которому он обучается. Обучающийся без учителя, самоорганизующийся слой Кохонена дает выходы в недетерминированных позициях. Они отображаются в желаемые выходы слоем Гроссберга.[3].

Таким образом, задача "идентификации" лиц успешно решается с помощью обученной сети встречного распространения. Другой актуальной задачей является распознавание машинописных и рукописных текстов в силу ее повседневной необходимости. К данной задаче также применим алгоритм, основанный на обучении сети встречного распространения.

Следовательно, сети встречного распространения включают в себя широкий круг задач по распознаванию, восстановлению и классификации образов. По своим возможностям сети встречного распространения превосходят возможности однослойных сетей. Время же их обучения может уменьшаться в сто раз. Обладая собственными интересными и полезными свойствами, сети встречного распространения могут давать решения в тех случаях, где долгая обучающая процедура невозможна. Уникальная способность сети встречного распространения - порождать функцию и обратную к ней - делает эту сеть полезной в ряде приложений. К преимуществам сети встречного распространения можно отнести и то, что она достаточно проста. Она дает возможность извлекать статистические свойства из множеств входных сигналов. Сеть полезна для приложений, в которых требуется быстрая начальная аппроксимация.

Алгоритм построения сетевой модели на основе начальных данных задачи сетевого планирования

Шорникова О.С.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ им. В.И. Вернадского
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА ИНФОРМАТИКИ (ГРУППА 500-И)

Системы сетевого планирования и управления, использующие сетевые модели, широко применяются в сферах деятельности, где необходим чёткий логический вывод, точный расчёт времени. Системы сетевого планирования и управления решают такие проблемы, как получение достоверной информации о состоянии дел, о возникших задержках и возможностях ускорения хода работ, концентрируют внимание руководителей на „критических“ работах, определяющих продолжительность проекта в целом. В результате это даёт возможность совершенствовать технологию и организацию работ, непосредственно влияющих на сроки проведения разработки, помогают составлять рациональные планы работ, обеспечивают согласованность действий исполнителей. В отсутствие сетевого графика поставленные задачи были бы слишком трудоёмкими и требовали бы участие многих исполнителей. Необходимость параллельного выполнения работ, зависимость начала многих работ от результатов других, значительно осложняют планирование разработки. Поэтому сетевое планирование и управление стало немаловажной частью всей структуры нынешних предприятий и экономики в целом. Отметим также, что методы сетевого планирования и управления удобно использовать в планировании и разработке сложных программных комплексов, в которых определены различные процессы, их взаимосвязь и последовательность.

Сетевая модель представляет собой план выполнения некоторого комплекса взаимосвязанных работ (операций), заданного в специфической форме сети, графическое изображение которой называется сетевым графиком.

Сетевая модель может быть задана различными видами начальной информации:

- 1.) списком работ проекта $\{A_i\}$, а также списком событий $\{i\}$, который необходим для определения связей, представленных в виде таблицы;
- 2.) диаграммой Ганта - горизонтальной линейной диаграммой, на которой задачи проекта представляются протяженными во времени отрезками, характеризующимися датами начала, окончания и задержками в установленном порядке следования.

На основе таких входных данных строится сетевая модель, предполагающая задание списков $\{A_j\}$ - имён процессов (работ) проекта, $\{t_j\}$ - времени выполнения работ, $R_j = \{A_k\}$ - предшествующих работ.

В настоящей статье обсуждается алгоритм построения сетевой модели и её графическая реализация по имеющимся начальным данным.

Отличительной особенностью сетевой модели является четкое определение всех временных взаимосвязей предстоящих работ. Главными элементами сетевой модели являются работы и события.

Работа определяется как:

- 1.) *действие* - протяженный во времени процесс, не требующий затрат ресурсов;
- 2.) *ожидание* - протяженный во времени процесс, не требующий затрат труда;
- 3.) *зависимость*, или фиктивная работа - логическая связь между двумя или несколькими работами (событиями), не требующими затрат труда, материальных ресурсов или времени. Оно указывает, что возможность одной работы непосредственно зависит от результатов другой. Причём продолжительность фиктивной работы принимается равной нулю.

Событие - это момент завершения какого - либо процесса, отражающий отдельный этап выполнения проекта. Событие может являться частным результатом отдельной работы или суммарным результатом нескольких работ. Событие может свершиться только тогда, когда закончатся все работы, ему предшествующие. Последующие работы могут начаться только тогда, когда событие совершится.

Алгоритм построения сетевого графика должен обеспечивать построение сетевой модели, которая не противоречит следующим правилам:

Правило 1. Каждый процесс в проекте представим одной и только одной работой - дугой графа.

Правило 2. Каждый процесс определяется двумя концевыми узлами. С помощью фиктивного процесса можно представить два параллельных процесса A_i и B_j . По определению фиктивный процесс не использует временных или других ресурсов. Вставив фиктивный процесс мы определяем процессы A_i и B_j одним концевым узлом (как требует правило 2).

Правило 3. В сетевой модели не должно быть "тушиковых" событий, т.е. событий, из которых не выходит ни одна работа, за исключением завершающего события.

Правило 4. В сетевом графике не должно быть "хвостовых" событий (кроме исходного), которым не предшествует хотя бы одна работа.

Правило 5. В сети рекомендуется иметь одно исходное и одно завершающее событие, за исключением случая неполной зависимости работ.

Сетевой алгоритм должен упорядочивать сетевой график. Упорядочение сетевого графика заключается в таком расположении событий и работ, при котором для любой работы предшествующее ей событие имеет меньший номер по сравнению с завершающим эту работу событием. То есть в упорядоченном сетевом графике все работы - дуги направлены слева направо: от событий с меньшими номерами к событиям с большими номерами.

Сетевая модель и соответствующий ей сетевой график могут быть построены вручную, непосредственным представлением событий и работ на основе исходных данных. Однако для более быстрого и удобного построения необходим алгоритм, который автоматически построит сетевую модель. Этот алгоритм использует три списка: $\{A_j\}$ - список имен процессов (работ); $\{t_j\}$ - список времени выполнения работ; $\{R_j\} = \{\{A_k\}\}$ - список предшествующих работ; ($j = \overline{1, n}$ - номера работ). На выходе формируется два списка: $\{S_i\}$ - список событий; $\{t_{ij}\}$ - список работ; (i - номера начала работ, j - номера окончания работ).

Далее приведён текст алгоритма, который реализует построение сетевой модели на основе заданных списков:

Шаг 1. Начало проекта; номер события $i := 0$; Положим $j := 1$

Шаг 2. Для работы A_j с временем выполнения t_j просматриваем список $\{R_j\} = \{A_k\}$.

1. Если $\{R_j\} = \{0\}$, то положить $(t_{ij} := t_j; j := j + 1)$
2. Если $\{R_j\} = \{p\}$, то положить $(i := \{p\}; t_{ij} = t_j; j = j + 1)$
3. Если $\{R_j\} = \{i_1, \dots, i_l\}$, то найти $p = \operatorname{argmax}\{t_{ik} | 1 \leq k \leq l\}$ и положить $(i := p; t_{ij} := t_j; (\forall \alpha = \overline{1, l}, \alpha \neq p) : t_{\alpha j} := 0); i := i + 1$

Шаг 3. Если $i = n$ алгоритм останавливается.

Представленный выше алгоритм построения сетевого графика является частью магистерской работы, в которой также представлены программные реализации Алгоритма построения остоного дерева, Алгоритма нахождения кратчайшего пути от заданной до любой другой вершины (Алгоритм Дейкстры), Алгоритма нахождения кратчайшего пути между двумя любыми вершинами (Алгоритм Флойда). Эти алгоритмы в целом составляют программный комплекс для решения задач, основанных на сетевых моделях.

ЛИТЕРАТУРА

- [1] Хэмди А. Таха. *Введение в исследование операций*. // 6-е издание. Пер. с англ. – 2001.

- [2] Н.Ш.Кремер, Б.А.Путко, И.М.Тришин, М.Н.Фридман. *Исследование операций в экономике.* // 2000. – 407с.
- [3] Свами М., Тхуласираман К. *Графы, сети и алгоритмы.* // 1984. – 454 с.

Неприводимые представления четверки рефлексий в кватернионном гильбертовом бимодуле

Щербина Ю.С.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА АЛГЕБРЫ И ФУНКЦИОНАЛЬНОГО АНАЛИЗА (ГРУППА 500)

4. ПОСТАНОВКА ЗАДАЧИ.

В алгебре $L[H]$ ограниченных линейных операторов над гильбертовым \mathbb{H} -бимодулем H самостоятельный интерес представляет задача описания множества вещественных параметров β_t , $t = \overline{1, m}$, для которых существует такой бимодуль H и m рефлексий R_1, \dots, R_m в H таких, что $\sum_{t=1}^m R_t = 0$. Напомним, что алгебра кватернионов \mathbb{H} — это некоммутативная ассоциативная \mathbb{R} -алгебра с делением размерности 4, базисные единицы $1, i, j, k$ которой удовлетворяют следующим правилам умножения: $i^2 = j^2 = k^2 = -1$, $ij = -ji = k$. При этом всякий кватернион q можно записать следующим образом: $q = q_0 + q_1i + q_2j + q_3k$, $q_t \in \mathbb{R}$, $t = \overline{0, 3}$.

Определение 23. Оператор $R \in L[H]$ называется рефлексией, если $R^* = -R$ и $R^2 = -I$.

В настоящей работе решается задача описания всех неприводимых четверок рефлексий R_t , $t = \overline{1, 4}$, действующих в кватернионном гильбертовом бимодуле H конечной размерности n , таких, что

$$\sum_{t=1}^4 \beta_t R_t = 0, \quad \beta_t \geq 0.$$

5. КОМПЛЕКСИФИКАЦИЯ ЗАДАЧИ.

Поставленная задача может быть сформулирована в терминах теории представлений $*$ -алгебры: а именно, описать все неприводимые представления $*$ -алгебры

$$\mathcal{R}_{4,\beta} = \mathbb{R} \langle r_1, r_2, r_3, r_4 \mid r_t^* = -r_t, r_t^2 = -e, t = \overline{1, 4} \rangle.$$

над конечномерными гильбертовыми кватернионными бимодулями.

Для решения этой задачи перейдем к комплексификации $H^{\mathbb{C}}$ кватернионного бимодуля H . В этом случае симплектические образы R_t^s операторов R_t , $t = \overline{1,4}$, связаны аналогичным равенством

$$\sum_{t=1}^4 \beta_t R_t^s = 0. \quad (1)$$

В соответствии со структурой рефлексии для всякого $t = \overline{1,4}$ имеет место разложение

$$H^{\mathbb{C}} = H_t^i \oplus_{\mathbb{C}} H_t^{-i},$$

где H_t^i, H_t^{-i} — \mathbb{C} -подмодули, соответствующие собственным значениям $i, -i$ оператора R_t^s . Так как $H_t^{-i} = R_j H_t^i$, то

$$\dim [H_t : \mathbb{C}] = \frac{1}{2} \dim [H^{\mathbb{C}} : \mathbb{C}] = \dim [H : \mathbb{H}].$$

Следовательно, симплектические образы R_t^s , $t = \overline{1,4}$, имеют следующие спектральные разложения:

$$R_t^s = i P_t(i) - i P_t(-i), \quad (2)$$

где ортопроекторы $P_t(i), P_t(-i)$ на соответствующие собственные подмодули для каждого $t = \overline{1,4}$ являются разложениями единицы. В частности, $P_t(-i) = I - P_t(i)$. Отсюда $-i R_t^s = 2P_t(i) - I$, и равенство (1) можно переписать следующим образом:

$$\sum_{t=1}^4 2\beta_t P_t(i) = \beta I, \quad \beta = \sum_{t=1}^4 \beta_t. \quad (3)$$

Обозначив через $\alpha_t = \frac{2\beta_t}{\beta}$, из равенства (3) окончательно получим условие на семейство ортопроекторов над \mathbb{C} -модулем $H^{\mathbb{C}}$:

$$\sum_{t=1}^4 \alpha_t P_t = I, \quad \sum_{t=1}^4 \alpha_t = 2. \quad (4)$$

Таким образом, мы свели исходную задачу к описанию (с точностью до унитарной эквивалентности) всех неприводимых четверок ортопроекторов с условием (4), действующих в комплексном модуле.

6. ПРЕДСТАВЛЕНИЯ * - АЛГЕБРЫ $P_{4,f}$.

Полученная задача реализуется как представления следующей *-алгебры над полем \mathbb{C} :

$$P_{4,f} = \mathbb{C} \left\langle p_1, p_2, p_3, p_4 \mid p_t = p_t^* = p_t^2, \sum_{t=1}^4 \alpha_t p_t = e \right\rangle. \quad (5)$$

Неприводимые представления этой алгебры описаны в работе В.Кругляка [2]. Согласно этим результатам, неприводимые представления *- алгебры $P_{4,f}$ всегда имеют размерность 2 и матрицы операторов R_t^s , $t = \overline{1,4}$ могут быть представлены следующими параметрическими формулами:

$$\begin{aligned} A. R_1^s &= \begin{pmatrix} 2i\lambda & i\sqrt{1-4\lambda^2} \\ i\sqrt{1-4\lambda^2} & -2i\lambda \end{pmatrix}, R_2^s = \begin{pmatrix} -2i\lambda & ie^{i\chi}\sqrt{1-4\lambda^2} \\ ie^{-i\chi}\sqrt{1-4\lambda^2} & 2i\lambda \end{pmatrix}, \\ R_3^s &= \begin{pmatrix} -2i\lambda & -ie^{-i\chi}\sqrt{1-4\lambda^2} \\ -ie^{-i\chi}\sqrt{1-4\lambda^2} & 2i\lambda \end{pmatrix}, R_4^s = \begin{pmatrix} 2i\lambda & -i\sqrt{1-4\lambda^2} \\ i\sqrt{1-4\lambda^2} & -2i\lambda \end{pmatrix}, \\ B. R_1^s &= \frac{1}{2\alpha_1\lambda} \times \\ & \begin{pmatrix} i(2\lambda^2 - \frac{1}{2}(\alpha_4^2 - \alpha_1^2)) & i\sqrt{-4\lambda^4 + 2(\alpha_4^2 + \alpha_1^2)\lambda^2 - \frac{1}{4}(\alpha_4^2 - \alpha_1^2)^2} \\ i\sqrt{-4\lambda^4 + 2(\alpha_4^2 + \alpha_1^2)\lambda^2 - \frac{1}{4}(\alpha_4^2 - \alpha_1^2)^2} & i(-2\lambda^2 + \frac{1}{2}(\alpha_4^2 - \alpha_1^2)) \end{pmatrix}, \\ R_2^s &= \frac{1}{2\alpha_2\lambda} \times \\ & \begin{pmatrix} i(-2\lambda^2 + \frac{1}{2}(\alpha_3^2 - \alpha_2^2)) & ie^{i\chi}\sqrt{-4\lambda^4 + 2(\alpha_2^2 + \alpha_3^2)\lambda^2 - \frac{1}{4}(\alpha_3^2 - \alpha_2^2)^2} \\ ie^{-i\chi}\sqrt{-4\lambda^4 + 2(\alpha_2^2 + \alpha_3^2)\lambda^2 - \frac{1}{4}(\alpha_3^2 - \alpha_2^2)^2} & i(2\lambda^2 - \frac{1}{2}(\alpha_3^2 - \alpha_2^2)) \end{pmatrix}, \\ R_3^s &= \frac{1}{2\alpha_3\lambda} \times \\ & \begin{pmatrix} i(-2\lambda^2 - \frac{1}{2}(\alpha_3^2 - \alpha_2^2)) & -ie^{-i\chi}\sqrt{-4\lambda^4 + 2(\alpha_2^2 + \alpha_3^2)\lambda^2 - \frac{1}{4}(\alpha_3^2 - \alpha_2^2)^2} \\ -ie^{-i\chi}\sqrt{-4\lambda^4 + 2(\alpha_2^2 + \alpha_3^2)\lambda^2 - \frac{1}{4}(\alpha_3^2 - \alpha_2^2)^2} & i(2\lambda^2 + \frac{1}{2}(\alpha_3^2 - \alpha_2^2)) \end{pmatrix}, \\ R_4^s &= \frac{1}{2\alpha_4\lambda} \times \\ & \begin{pmatrix} i(2\lambda^2 + \frac{1}{2}(\alpha_4^2 - \alpha_1^2)) & -i\sqrt{-4\lambda^4 + 2(\alpha_4^2 + \alpha_1^2)\lambda^2 - \frac{1}{4}(\alpha_4^2 - \alpha_1^2)^2} \\ -i\sqrt{-4\lambda^4 + 2(\alpha_4^2 + \alpha_1^2)\lambda^2 - \frac{1}{4}(\alpha_4^2 - \alpha_1^2)^2} & i(-2\lambda^2 - \frac{1}{2}(\alpha_4^2 - \alpha_1^2)) \end{pmatrix}, \end{aligned}$$

где $0 \leq \lambda < \frac{1}{2}$, $\mu > 0, \nu \in \mathbb{R}$. При этом $0 \leq \chi < \frac{\pi}{2}$, когда $\lambda = 0$, и $-\frac{\pi}{2} \leq \chi < \frac{\pi}{2}$, когда $0 < \lambda < \frac{1}{2}$.

Анализ матриц полученных операторов позволяет сделать вывод, что каждый из этих операторов можно рассматривать как симплектический образ соответствующей рефлексии. Так как $\dim[H : \mathbb{H}] = \frac{1}{2} \dim[H^{\mathbb{C}} : \mathbb{C}] = 1$, то в случае А общий вид неприводимой четверки рефлексий R_t , $t = \overline{1,4}$, действующей в кватернионном гильбертовом бимодуле H размерности 1 следующий:

$$\begin{aligned} R_1 &= 2i\lambda + \sqrt{1-4\lambda^2}k, & R_2 &= -2i\lambda + e^{-i\chi}\sqrt{1-4\lambda^2}k, \\ R_3 &= -2i\lambda - e^{i\chi}\sqrt{1-4\lambda^2}k, & R_4 &= 2i\lambda - k\sqrt{1-4\lambda^2}k. \end{aligned}$$

$$\begin{aligned}
B. R_1 &= \frac{1}{2\alpha_1\lambda} \left(i \left(2\lambda^2 - \frac{1}{2}(\alpha_4^2 - \alpha_1^2) \right) + \sqrt{-4\lambda^4 + 2(\alpha_4^2 + \alpha_1^2)\lambda^2 - \frac{1}{4}(\alpha_4^2 - \alpha_1^2)^2 k} \right), \\
R_2 &= \frac{1}{2\alpha_2\lambda} \left(i \left(-2\lambda^2 + \frac{1}{2}(\alpha_3^2 - \alpha_2^2) \right) + e^{i\chi} \sqrt{-4\lambda^4 + 2(\alpha_2^2 + \alpha_3^2)\lambda^2 - \frac{1}{4}(\alpha_3^2 - \alpha_2^2)^2 k} \right), \\
R_3 &= \frac{1}{2\alpha_3\lambda} \left(i \left(-2\lambda^2 - \frac{1}{2}(\alpha_3^2 - \alpha_2^2) \right) - e^{-i\chi} \sqrt{-4\lambda^4 + 2(\alpha_2^2 + \alpha_3^2)\lambda^2 - \frac{1}{4}(\alpha_3^2 - \alpha_2^2)^2 k} \right), \\
R_4 &= \frac{1}{2\alpha_4\lambda} \left(i \left(2\lambda^2 + \frac{1}{2}(\alpha_4^2 - \alpha_1^2) \right) - \sqrt{-4\lambda^4 + 2(\alpha_4^2 + \alpha_1^2)\lambda^2 - \frac{1}{4}(\alpha_4^2 - \alpha_1^2)^2 k} \right),
\end{aligned}$$

где $0 \leq \lambda < \frac{1}{2}$, $\mu > 0, \nu \in \mathbb{R}$. При этом $0 \leq \chi < \frac{\pi}{2}$, когда $\lambda = 0$, и $-\frac{\pi}{2} \leq \chi < \frac{\pi}{2}$, когда $0 < \lambda < \frac{1}{2}$.

Заметим, что, вообще говоря, из неприводимости представления алгебры над \mathbb{H} -модулем H не следует неприводимость этого представления над \mathbb{C} -модулем $H^{\mathbb{C}}$. Однако в нашем случае это справедливо. Действительно, если предположить, что семейство операторов $R_t^s, t = \overline{1,4}$, приводимо над \mathbb{C} -модулем $H^{\mathbb{C}}$, то существует нетривиальный минимальный инвариантный \mathbb{C} -подмодуль $H_1^{\mathbb{C}}$, размерность которого равна 2, причем, согласно приведенным выше результатам, операторы $R_t^s, t = \overline{1,4}$, являются симплектическими образами \mathbb{H} -линейных операторов в \mathbb{H} -подмодуле $H_1, \dim[H_1 : \mathbb{H}] = 1$. Следовательно, исходное семейство операторов $R_t, t = \overline{1,4}$, также приводимо над \mathbb{H} -модулем H .

ЛИТЕРАТУРА

- [1] Galinsky D.V., Kruglyak S.A. *Representations of $*$ -algebras generated by linearly related orthogonal projections.* // (Ukrainian.) Visnyk Kyiv. derzh. Univ.No - 2, 24 - 31. - 1999.
- [2] Kruglyak S.A., Nazarova L.A., Roiter A.V. *On Regular Locally Scalar Representations of Graph \widetilde{D}_4 in Hilbert Spaces.* // Укр.мат.журнал. - 2003. - т.55 - №.9. - С.1277-1283.
- [3] Kruglyak S.A. *Coxeter functors for a certain class of $*$ -quivers and $*$ -algebras.* *Methods Funct.* // Anal. Topology 8 (2002), no. 4. 49-57.

Автоматный подход к решению систем линейных однородных диофантовых уравнений в области $\{0,1\}$

Ющенко С.В.

ТАВРИЧЕСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМ. В.И. ВЕРНАДСКОГО
ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАТИКИ
КАФЕДРА АЛГЕБРЫ И ФУНКЦИОНАЛЬНОГО АНАЛИЗА (ГРУППА 500)

Предлагается автоматный подход реализации алгоритма построения базиса множества решений СЛОДУ в области $\{0,1\}$.

Алгоритмы построения базиса множества всех решений системы линейных однородных диофантовых уравнений (СЛОДУ) во множестве натуральных чисел встречаются во многих приложениях. Известны следующие методы решения СЛОДУ: метод Гаусса, метод Контежан-Дэви, метод Потье, не обладающие свойством инкрементальности и алгоритм построения TSS, алгоритм Доменжуда, которые этим свойством обладают, что позволяет использовать эти алгоритмы и для решения СЛНДУ, систем линейных неравенств.

При этом TSS-алгоритм возможно дает минимальное порождающее множество решений СЛОДУ. Построение для заданной СЛОДУ базиса множества всех решений будем проводить с помощью оптимального алгоритма, основанного на идее TSS-метода [1]. Данный алгоритм требует введения только двух дополнительных переменных, что делает его достаточно экономным по памяти. Рассмотрим СЛОДУ, решения которых ищутся во множестве $\{0,1\}$, а коэффициенты берутся из множества $\{-1,0,1\}$. Пусть $\mathbf{A} = (A, X, f, \{a_0\}, F)$ — конечный автомат без выходов, где A — конечное множество состояний, X — конечное множество входных символов, $f : A \times X \rightarrow A$ — (частичная) функция переходов, $a_0 \in A$ — начальное состояние, $F \subseteq A$ — множество заключительных состояний.

Для уравнение вида:

$$a_1x_1 + a_2x_2 + \dots + a_qx_q = 0, \tag{1}$$

$$a_j \in \{-1, 0, 1\}, x_j \in \{0, 1\}, j \in [1, q]$$

Применим оптимальный TSS-алгоритм и получим базис множества решений уравнения(1). Представим этот базис с помощью автомата \mathbf{A} , где множество состояний автомата $A = \{0, 1, -1\}$, $X = \{e_1, e_2, \dots, e_q\}$, $e_i (i \in [1, q])$ — вектора канонического базиса множества N^q , начальным и заключительным состоянием является состояние 0, а функция переходов f определяется так:

$$\begin{aligned} f(0, e_i) &= 0, \text{ если } e_i \text{ соответствует коэффициенту } 0; \\ f(0, e_i) &= 1, \text{ если } e_i \text{ соответствует коэффициенту } 1; \end{aligned}$$

$$\begin{aligned}
 f(0, e_i) &= -1, \text{ если } e_i \text{ соответствует коэффициенту } -1; \\
 f(1, e_i) &= 0, \text{ если } e_i \text{ соответствует коэффициенту } -1; \\
 f(-1, e_i) &= 1, \text{ если } e_i \text{ соответствует коэффициенту } 1;
 \end{aligned}$$

иначе функция f неопределена. Рассмотрим СЛОДУ

$$\begin{cases} 1x_1 + 0x_2 - 1x_3 - 1x_4 + 1x_5 + 1x_6 - 1x_7 = 0 \\ 1x_1 - 1x_2 + 0x_3 + 0x_4 - 1x_5 - 1x_6 + 1x_7 = 0 \end{cases}$$

имеем: $A = \{0, 1, -1\}$, $X = \{e_1, e_2, \dots, e_7\}$, $a_0 = 0$, $F = \{a_0\}$, а функции переходов f определяются такой таблицей переходов:

Для первого уравнения				Для второго уравнения			
f	0	1	-1	f_1	0	1	-1
e_1	1	-	0	e_1	1	-	0
e_2	0	-	-	e_2	-1	0	-
e_3	-1	0	-	e_3	0	-	-
e_4	-1	0	-	e_4	0	-	-
e_5	1	-	0	e_5	-1	0	-
e_6	1	-	0	e_6	-1	0	-
e_7	-1	0	-	e_7	1	-	0

Таблица переходов прямого произведения полученных автоматов после удаления недостижимых и тупиковых состояний выглядит следующим образом:

f_2	(0,0)	(0,-1)	(-1,0)	(1,-1)	(-1,1)	(1,0)	(0,1)
e_1	-	(1,0)	(0,1)	-	-	-	-
e_2	(0,-1)	-	-	-	-	(0,0)	(0,0)
e_3	(-1,0)	-	-	-	-	-	-
e_4	(-1,0)	-	-	-	-	-	-
e_5	(1,-1)	-	(0,-1)	-	(0,0)	(1,0)	(1,0)
e_6	(1,-1)	-	(0,-1)	-	(0,0)	(1,0)	(1,0)
e_7	(-1,1)	(-1,0)	-	(0,0)	-	-	-

Начальным и заключительным состоянием данного автомата является состояние (0,0). Поиск простых циклов в полученном автомате приводит к таким словам:

$$p_1 = e_2e_1e_3, p_2 = e_2e_1e_4, p_3 = e_3e_1e_5e_4, p_4 = e_3e_1e_6e_4, p_5 = e_5e_7,$$

$$p_6 = e_6e_7, p_7 = e_7e_6, p_8 = e_2e_7e_1e_5e_3, p_9 = e_2e_7e_1e_6e_3$$

и так далее. Эти слова порождают следующие базисные решения:

$$s_1 = (1, 1, 1, 0, 0, 0, 0), s_2 = (1, 1, 0, 1, 0, 0, 0), s_3 = (1, 0, 1, 1, 1, 0, 0),$$

$$s_4 = (1, 0, 1, 1, 0, 1, 0), s_5 = (0, 0, 0, 0, 1, 0, 1), s_6 = (0, 0, 0, 0, 0, 1, 1).$$

Не ограничивая общности, рассмотрим СЛОДУ с двумя уравнениями. Пусть x — произвольное решение СЛОДУ, тогда вектору x в автомате, представляющем первое уравнение, соответствует слово p и этому вектору x соответствует такое же слово p и в автомате, представляющем второе уравнение данной СЛОДУ. Это вытекает из того, что исходные автоматы представляют множество всех решений СЛОДУ. Но тогда слово p , будучи элементом пересечения двух регулярных языков, представленным в прямом произведении исходных автоматов, должно переводить этот автомат из начального состояния в начальное состояние. Это значит, что слову p соответствует некоторый цикл в этом автомате. Представим это слово, как конкатенацию его подслов p_1, p_2, \dots, p_k , соответствующих простым циклам. Поскольку слова p_1, p_2, \dots, p_k генерируют базисные решения, то пусть это будут решения e_1, e_2, \dots, e_k . Тогда становится очевидным, что $x = e_1 + e_2 + \dots + e_k$.

Следовательно, имеет место следующая теорема [2]:

Теорема 14. *Множество решений СЛОДУ, соответствующих словам, представляющим простые циклы в автомате, который есть прямым произведением исходных автоматов, содержит все базисные решения данной СЛОДУ.*

ЛИТЕРАТУРА

- [1] Krivoi S. Об алгоритмах решения систем линейных диофантовых констрейнтов в области $\{0,1\}$, ж. Кибернетика и системный анализ. 2003, № 5. с. 58-69.
- [2] Krivoi S., Matveeva L., Gjivach V. Автоматный метод решения систем линейных ограничений в области $\{0,1\}$, KDS 2005 4.4. Intelligent Technologies in Control, p.389-393.